RC71xx

AT Command Reference

SIERRA WIRELESS®

41114675 Rev. 3 January 2025

Important Notice

Information relating to this product and the application or design described herein is believed to be reliable, however such information is provided as a guide only and Semtech assumes no liability for any errors in this document, or for the application or design described herein.

Semtech reserves the right to make changes to the product or this document at any time without notice. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. Semtech warrants performance of its products to the specifications applicable at the time of sale, and all sales are made in accordance with Semtech's standard terms and conditions of sale.

SEMTECH PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS, OR IN NUCLEAR APPLICATIONS IN WHICH THE FAILURE COULD BE REASONABLY EXPECTED TO RESULT IN PERSONAL INJURY, LOSS OF LIFE OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. INCLUSION OF SEMTECH PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE UNDERTAKEN SOLELY AT THE CUSTOMER'S OWN RISK. Should a customer purchase or use Semtech products for any such unauthorized application, the customer shall indemnify and hold Semtech and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs damages and attorney fees which could arise.

Wireless Communications

Due to the nature of wireless communications, transmission and reception of data can never be guaranteed. Data may be delayed, corrupted (i.e., have errors) or be totally lost. The Semtech product should not be used in situations where failure to transmit or receive data could result in damage of any kind to the user or any other party, including but not limited to personal injury, death, or loss of property. Semtech accepts no responsibility for damages of any kind resulting from delays or errors in data transmitted or received using the Semtech product, or for failure of the Semtech product to transmit or receive such data.

Safety

Do not operate the Semtech product in areas where blasting is in progress, where explosive atmospheres may be present, near medical equipment, near life support equipment, or near any equipment which may be susceptible to any form of radio interference. In such areas, the Semtech product should be powered off.

Qualcomm licenses

Semtech's cellular modules are sold subject to certain notices and restrictions regarding patent licenses from Qualcomm Incorporated. These notices and restrictions are available at www.sierrawireless.com/qualcomm-notices.

Sierra Wireless

Semtech Corporation acquired Sierra Wireless in January 2023. The Sierra Wireless brand is gradually being phased out. During the phase-out period, references to both "Semtech" and "Sierra Wireless" may appear in product documentation.

Contact Information

Sales information and technical support, including warranty and returns	Web: sierrawireless.com/company/contact-us/ Global toll-free number: 1-877-687-7795 6:00 am to 5:00 pm PST
Corporate and product information	Web: sierrawireless.com

Revision History

Revision number	Release date	Changes
1	Jan 2024	General release
2	May 2024	• Updated +CCHC (response format), +CFUN (added <fun>=4), +CGDCONT (<cid> range), +KCERTSTORE (password required; example responses), +KPRIVKSTORE (example responses), +KTCPCFG (response format — error), +KUDPCFG (added <cipher_suite>; updated <mode>, <udp_notif>, <data_mode>), +WESHDOWN (<gpio_index> values) • Updated +CGPADDR (renamed from +KCGPADDR)
3	January 2025	• Added !WIFISCAN

Contents

1: About This Guide

1.1 Introduction

This document describes supported standard and proprietary AT commands available for Sierra Wireless RC71xx products, and provides details where commands vary from the standards.

These commands are intended for use by OEMs, and are supplemental to the standard AT commands for GSM devices defined by the 3GPP (3rd Generation Partnership Project) in *TS 27.007 AT command set for User Equipment (UE)* and *TS 27.005 Use of Data Terminal Equipment—Data Circuit terminating Equipment (DTE-DCE) interface for Short Message Service (SMS) and Cell Broadcast Service (BSE)*.

Note: For questions or concerns relating to command implementation, please contact your Sierra Wireless account representative.

1.2 Command access

Some commands in this reference are password-protected. To use these commands, enter the correct password using AT!ENTERCND on page 12. Once the password is entered, all commands are available and remain available until the modem is reset or powered off and on.

The password assigned to !ENTERCND is unique to each customer and is configured onto the modem during manufacture. If you do not know your password, contact your Sierra Wireless Account Manager or Sierra Wireless distributor.

1.3 Command timing

1.3.1 Interval timing

Some commands require time to process before additional commands are entered.

When building automated test scripts, ensure that sufficient delays are embedded, where necessary, to avoid these errors.

1.3.2 Escape sequence guard time

The AT escape sequence "**+++**" requires a guard time of 1.0 seconds before and after it is used.

1.4 Result codes

Result codes are not shown in the command tables unless special conditions apply. Generally the result code OK is returned when the command has been executed. ERROR may be returned if parameters are out of range, and is returned if the command is not recognized or is not permitted in the current state or condition of the modem.

1.5 Response formats

Response formats shown in this document are intended to accurately describe the non-whitespace content of responses. For display purposes within this document, extraneous whitespace content (blank lines between lines of text) may not be displayed, and whitespace (blank spaces) between text segments within lines may be shorter or longer than what is received in actual responses.

For example:

```
AT!THISEXAMPLE?

THISEXAMPLE:

TestVal1=7              TestVal2=Hello

OK
```

could be shown in this document without extra blank lines and with less space between TestVal1 and TestVal2

```
AT!THISEXAMPLE?

THISEXAMPLE:
TestVal1=7   TestVa2=Hello
OK
```

If automated scripts are used to parse command responses, make sure to parse whitespace appropriately.

1.6 References

This guide covers the command sets used by OEMs, designers and testers of Sierra Wireless AirPrime products, plus general operational use commands.

For additional product-specific documentation, refer to source.sierrawireless.com.

1.7 Terminology and acronyms

This document makes wide use of acronyms that are in common use in data communications and cellular technology.

1.8 Current firmware versions

1.8.1 Version

To determine your firmware revision, enter the identification command **AT+GMR**.

1.8.2 Upgrading

To check for newer modem firmware, go to the device page at source.sierrawireless.com and select the Firmware option.

1.9 Document structure

This document describes the proprietary commands listed in the tables below—each table corresponds to a specific chapter.

AT Password Commands—Commands used to enable access to password-protected AT commands and to set the AT command password.

Modem Status, Customization, and Reset Commands—Commands used to determine modem status, adjust customization settings, and reset the modem.

Diagnostic Commands—Commands used to select frequency bands and diagnose problems.

Test Commands—Commands required to place the modem in particular modes of operation, test host connectivity, and to configure the transmitters and receivers for test measurements.

Memory Management Commands—Commands that control the data stored in non-volatile memory of the modem.

SIM Commands—Commands used to communicate with an installed SIM.

I/O Commands—Commands used to configure and manage GPIOs, ADCs and other IOs.

Delta OTA Firmware Update Commands—Commands used for firmware updates.

Protocol Commands—Internet Protocol-related Commands.

Unsolicited Message Commands—Commands that are related to USL.

Supported GSM/WCDMA AT Commands—Commands that are supported by most Sierra Wireless devices.

Band Definitions—Commands include input and/or output ‘band’ parameters, where the value is an enumerated value representing a network technology and band or a 3GPP band number.

ASCII Table/CME Error Codes

1.10 Conventions

The following format conventions are used in this reference:

Character codes or keystrokes that are described with words or standard abbreviations are shown within angle brackets using a different font, such as <CR> for Carriage Return and <space> for a blank space character.

Numeric values are decimal unless prefixed as noted below.

Hexadecimal values are shown with a prefix of 0x, i.e. in the form 0x3D.

Binary values are shown with a prefix of 0b, i.e. in the form 0b00111101.

Command and register syntax is noted using an alternate font: **!CHAN=<c>[,b]**. The leading "**AT**" characters are not shown but must be included before all commands except as noted in the reference tables.

Characters that are required are shown in uppercase; parameters are noted in lowercase. Required parameters are enclosed in angle brackets (**<n>**) while optional parameters are enclosed within square brackets (**[x]**). The brackets are not to be included in the command string.

Commands are presented in table format. Each chapter covers the commands related to that subject and presents a summary table to help locate needed commands. Commands are in ASCII alphabetical order in the body of each chapter.

Any default settings are noted in the command tables. Note that these are the factory default settings and *not* the default parameter value assumed if no parameter is specified.

Result Code This is a numeric or text code that is returned after all commands (except resets)—text codes are returned if verbose responses are enabled. Only one result code is returned for a command line regardless of the number of individual commands contained on the line.

Response This term indicates a response from the modem that is issued prior to a result code. Reading registers or issuing commands that report information will provide a response followed by a result code unless the command generates an error.

Responses and result codes from the modem, or host system software prompts, are shown in this font:

```
CONNECT 14400
```

2: AT Password Commands

2.1 Introduction

AT commands described in this document are password-protected. This chapter describes how to enter and change the password.

2.2 Command summary

Table 2-1 on page 11 lists the commands described in this chapter.

Table 2-1: AT Password Commands

Command	Description	Page
!ENTERCND	Enable access to password-protected commands	12
!SETCND	Set AT command password	13

2.3 Command reference

Table 2-2: AT Password Command Details

Command	Description
!ENTERCND	**Enable access to password-protected commands** Before any password-protected AT commands can be used, !ENTERCND must be used to enter the password to gain access. The initial password is configured onto the modem during manufacture. You can change the password using !SETCND. If you do not know the password, contact your Sierra Wireless account manager. Once the password has been entered correctly, the password-protected AT commands are available until the modem is reset or powered off and on. **Password required:** Yes—Query format only. **Reset required to apply changes:** No **Persistent across power cycles:** No **Usage:** • Execution: **AT!ENTERCND=<"key">** Response: OK Purpose: Unlock password-protected commands. **Parameters:** <"key"> (Password stored in NV memory) • Length: 4–15 characters • Supported characters: '0'–'9', 'A'–'Z', 'a'–'z', special characters (e.g. "!#$%&'()*+,-./:<>=?@" Note: Double quotes (") are not allowed. • Characters may be entered in ASCII format, or in Hex format. (For example: "myPass3" or "ABCDEF01234".)

Table 2-2: AT Password Command Details (Continued)

Command	Description
!SETCND	**Set AT command password** Change the password used for the !ENTERCND command. (Before you can change the password using **!SETCND**, you must enable access to this command using **!ENTERCND**.) **Password required:** Yes (see !ENTERCND for details) **Reset required to apply changes:** No **Persistent across power cycles:** Yes **Usage:** • Execution: **AT!SETCND=<"key">** Response: OK *or* Password protected ERROR Purpose: Sets <"Key"> as the new password for accessing protected commands, or returns the "Password protected" error if a null key ("") is entered. **Parameters:** <"key"> (New password) • Length: 4–15 characters • Supported characters: '0'–'9', 'A'–'Z', 'a'–'z', special characters (e.g. "!#$%&'()*+,-./:<>=?@" Note: Double quotes (") are not allowed. • Characters may be entered in ASCII format, or in Hex format. (For example: "myPass3" or "ABCDEF01234".) • Note—This command does not allow a null password to be set.

3: Modem Status, Customization, and Reset Commands

3.1 Introduction

This chapter describes commands used to reset the modem, adjust customization settings, retrieve the hardware version, and monitor the temperature, voltage, and modem status.

3.2 Command summary

Table 3-1 lists the commands described in this chapter.

Table 3-1: Modem Status Commands

Command	Description	Page
+++	Switch from Data Mode to Command Mode	16
!ADC	Read ADC value	17
!ANTSEL	Set/query external antenna select configuration	18
!BAND	Set/return usable bands	19
+CEDRXRDP	Read eDRX Dynamic Parameters	20
+CEDRXS	Configure eDRX	21
+CESQ	Extended Signal Quality	22
+CGACT	Activate/deactivate PDP context	24
+CGAUTH	Set/Report PDP connection authentication parameters	25
+CGDCONT	Define PDP context	26
+CMEE	Report mobile termination error	28
+CPSMS	Configure Power Saving Mode (PSM)	29
+CPWROFF	Power Off	30
+CSQ	Display signal quality	31
!CUSTOM	Set/return customization settings	32
!GSTATUS	Return operational status	34
!HWID	Read hardware ID	37
I	Display product identification information	38
+KCELL	Display Detected Cell Details	39
+KGSN	Request Product Serial Number Identification and Software Version	40
+KSLEEP	Configure UART1 power management (sleep mode entry conditions)	42

Table 3-1: Modem Status Commands (Continued)

Command	Description	Page
+KSREP	Enable/disable startup reporting	43
+KSUP (notification)	Startup notification (unsolicited notification)	44
!NETDEVCTL	Bind/Unbind IP address with host for NIC USB interface	45
!NVBACKUP	Backup NV data	46
!PATEMP	Return PA temperature information	47
!PATEMP (notification)	PA temperature state change—Unsolicited notification	48
!PCINFO	Return power control status information	49
!PCTEMP	Return Power control temperature information	50
!PCTEMP (notification)	PMIC temperature state change—Unsolicited notification	51
!PCTEMPLIMITS	Set/report temperature state limit values	52
!PCVOLT	Return current power supply voltage information	53
!PCVOLT (notification)	PMIC voltage state change—Unsolicited notification	54
!PCVOLTLIMITS	Set/report power supply voltage state limit values	55
!POWERDOWN	Power down system	56
!POWERMODE	Enable/disable PSM	57
!POWERWAKE	Configure PSM wakeup sources	58
!PRIID	Report module PRI part number and revision	59
!RESET	Reset modem	60
!SCUMMTU	Report MTU Size	61
!SKU	Read Module SKU	62
!USBCOMP	Set/report USB interface configuration	63
!USBINFO	Return information from active USB descriptor	64
!USBPID	Set/report product ID in USB descriptor	65
+WESHDOWN	Emergency Shutdown	67
+WFWUPD	Download and install firmware delta package locally over AT port	68
+WFWUPD (notification)	Package install is launched—Unsolicited notification	70
!WIFISCAN	Scan for available Wi-Fi networks	71
+WUSLMSK	Enable/disable unsolicited notifications	74

3.3 Command reference

Table 3-2: Modem Status Command Details

Command	Description
+++	**Switch from Data Mode to Command Mode** **Notes:** • This command is only available during data mode. The +++ character sequence suspends the data flow over the AT interface and switches to command mode. This allows entering AT commands while maintaining the data connection to the remote device. • To return to data mode, use ATO[n]. • Line needs one second silence before and one second after (do not end with terminating character). • The +++ characters are not transmitted in the data flow. **Password required:** No **Usage:** • Execution: **+++** Response: OK **Parameters:** None

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!ADC	**Read ADC value** Return the reading for a specified ADC channel. **Password required:** No **Usage:** • Execution: **AT!ADC=<ADC Type>** Response: !ADC: <ADC Type>, <value> OK *or* ADC ERROR: PARAMETER ERROR *or* ADC ERROR: NOT SUPPORT *or* ERROR Purpose: Display the reading from the specified input source. *Note: AT!ADC just displays the raw value of ADC Sensor* • Query List: **AT!ADC=?** Purpose: Display valid execution format and parameter values. **Parameters:** <ADC Type> (ADC that is being read) • ASCII string • Valid options: • VBATT • PA_THERM • PMIC_THERM • ADC0 • ADC1 <value> (Value read from the ADC) • Unsigned integer

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!ANTSEL	**Set/query external antenna select configuration** Configure the modem to use ANT_CNTL0/ANT_CNTL1 to select the antenna to use for each specified LTE frequency band. (Any of the available pins that are not needed for a specific band should be configured as not required.) When the modem switches to a frequency band that has been configured using this command, the pins are driven as specified and the host uses them to tune the external antenna appropriately. **Notes:** When designing the system, and configuring the device: • Perform system level testing to ensure that the antenna switching feature does not introduce any handover issues. The tunable antenna should be designed to ensure that it can retune in < 5 µs (recommended) and < 10 µs (maximum). **Password required:** Yes (see !ENTERCND for details) **Reset required to apply changes:** Yes **Usage:** • Execution: **AT!ANTSEL=<band>, <ANT_CNTL0>, <ANT_CNTL1>** Response: OK Purpose: Configure the pins for the specified <band>. • Query: **AT!ANTSEL?** Response: BAND <band *a*>: <ANT_CNTL0>, <ANT_CNTL1> BAND <band *b*>: <ANT_CNTL0>, <ANT_CNTL1> ... OK Example: BAND 2: 1, 0 BAND 5: 1, 1 OK Purpose: Display the current external antenna select configuration. • Query List: **AT!ANTSEL=?** Purpose: Display valid execution format and parameter values. **Parameters:** <band> (RF band) • 3GPP band number. For a full listing of 3GPP band numbers, see Table 13-1 on page 210. • Valid range: 1–71. Band support is product specific—refer to the RC71xx Product Technical Specification document for details. <ANT_CNTL0>, <ANT_CNTRL1> (ANT_CNTL pin configurations) • 0—Logic low • 1—Logic high • 2—Not used for antenna selection

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!BAND *Note: The 'Basic' command and response versions are used if you haven't entered the required password. (See Command access on page 7.)*	**Set/return usable bands** Set or display the list of bands on which the modem can operate. **Password required:** Yes—Execution (Extended) format (see !ENTERCND for details) **Usage:** • Execution (Basic): **AT!BAND=0** Response: OK Purpose: Enable the modem to use all bands supported by the UE. • Execution (Extended): **AT!BAND=<band1>[,<band2>[,<band3>[...]]]** Response: OK Purpose: Enable the modem to use only the listed bands. • Query: **AT!BAND?** Response: !BAND: <band1>[,<band2>[,<band3>[...]]] OK Purpose: Report the current list of bands that the modem can use. • Query List: **AT!BAND=?** Response: !BAND: (<band1>,<band2>,...) OK Purpose: Display the list of all bands supported by the UE. **Parameters:** <band1>, <band2>, ... (Band number(s) on which the UE can operate) • Decimal • 3GPP band number **Examples:** at!band=? !BAND: (1,3,7,8,20,28) at!band=2 +CME ERROR: 308 ← *Fail (B2 is not supported)* at!band=1,3,2 +CME ERROR: 308 ← *Fail (B2 is not supported)* at!band=1,7,3 OK at!band? !BAND: 1,7,3 ← *Pass*

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+CEDRXRDP	**Read eDRX Dynamic Parameters** Read the current eDRX status and related parameters. **Notes:** • This implementation of +CEDRXRDP follows 3GPP TS 27.007, with exceptions as noted in the parameter descriptions. **Password required:** No **Usage:** • Execution: **AT+CEDRXRDP** Response: +CEDRXRDP: <AcT-type>[, <Requested_eDRX_value>[, <NW-provided_eDRX_value>[, <Paging_time_window>]]] OK Purpose: Report the current eDRX status and parameters. • Query List: **AT+CEDRXRDP=?** Purpose: Return the execution command format and the supported parameter values. **Parameters:** <AcT> (Relationship between Access technology Type (RAT) and requested eDRX value) • 0—RAT is not using eDRX • 1—EC-GSM-IoT (A/Gb mode) • 2—GSM (A/Gb mode) • 3—UTRAN (Iu mode) • 4—E-UTRAN (WB-S1 mode) • 5—E-UTRAN (NB-S1 mode) <Requested_eDRX_value> (eDRX value requested by module) 4 bits represented as a string. Refers to bits 4–1 of octet 3 of extended DRX parameters information element. For coding and value range details, refer to the +CEDRXRDP description in 3GPP TS 27.007. • 1101—Default <NW-provided_eDRX_value> (eDRX value provided by network) 4 bits represented as a string. Refers to bits 4–1 of octet 3 of extended DRX parameters information element. For coding and value range details, refer to the +CEDRXRDP description in 3GPP TS 27.007. • e.g. 0011 <Paging_time_window> (Paging time window length) 4 bits represented as a string. Refers to bits 8–5 of octet 3 of extended DRX parameters information element. For coding and value range details, refer to the +CEDRXRDP description in 3GPP TS 27.007. • e.g. 0001

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+CEDRXS	**Configure eDRX** Enable/disable eDRX and configure settings for specified RATs. **Notes:** • This implementation of +CEDRXS follows 3GPP TS 27.007, with exceptions as noted in the parameter descriptions. **Password required:** No **Usage:** • Execution: **AT+CEDRXS=<mode>[, <AcT-type>[, <Requested_eDRX_value>]]]** Response: OK *or* +CME ERROR: <err> Purpose: Enable/disable eDRX and configure setting for specified RAT. • Query: **AT+CEDRXS?** Response: +CEDRXS: <AcT-type>[, <Requested_eDRX_value> ... OK Purpose: Report current eDRX settings for each RAT that has eDRX enabled. • Query List: **AT+CEDRXS=?** Purpose: Return the execution command format and the supported parameter values. **Parameters:** <mode> (Enable/Disable LTE eDRX) • 0—Disable eDRX • 1—Enable eDRX • 2—Enable eDRX and enable the unsolicited result code +CEDRXP: • +CEDRXP: <AcT-type>[, <Requested_eDRX_value>[, <NW-provided_eDRX_value [, <Paging_time_window>]]] • 3—Disable eDRX, discard eDRX parameters and reset to default values <AcT> (Access technology Type (RAT) and relationship to requested eDRX value) • 0—RAT is not using eDRX • 1—EC-GSM-IoT (A/Gb mode) • 2—GSM (A/Gb mode) • 3—UTRAN (Iu mode) • 4—E-UTRAN (WB-S1 mode) • 5—E-UTRAN (NB-S1 mode) <Requested_eDRX_value> (eDRX value requested by module) 4 bits represented as a string. Refers to bits 4–1 of octet 3 of extended DRX parameters information element. For coding and value range details, refer to the +CEDRXRDP description in 3GPP TS 27.007. • 1101—Default <NW-provided_eDRX_value> (eDRX value provided by network) 4 bits represented as a string. Refers to bits 4–1 of octet 3 of extended DRX parameters information element. For coding and value range details, refer to the +CEDRXRDP description in 3GPP TS 27.007. • e.g. 0011 <Paging_time_window> (Paging time window length) 4 bits represented as a string. Refers to bits 8–5 of octet 3 of extended DRX parameters information element. For coding and value range details, refer to the +CEDRXRDP description in 3GPP TS 27.007. • e.g. 0001

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+CESQ	**Extended Signal Quality** **Notes:** • If the current serving cell is not a GERAN cell, <rxlev> and <ber> are set to value 99. • If the current serving cell is not a UTRA FDD or UTRA TDD cell, <rscp> is set to 255. • If the current serving cell is not a UTRA FDD cell, <ecno> is set to 255. • If the current serving cell is not an E-UTRA cell, <rsrq> and <rsrp> are set to 255 • Therefore, the RC71xx returns 99 for rxlev/ber, and 255 for rscp/ecno. **Password required:** No **Usage:** • Execution: **AT+CESQ** Response: +CESQ: <rxlev>, <ber>, <rscp>, <ecno>, <rsrq>, <rsrp> OK Purpose: Display signal quality parameters. • Query List: **AT+CESQ=?** Purpose: Display valid parameter values. **Parameters:** <rxlev> (Received signal strength level (see 3GPP TS 45.008 [20] subclause 8.1.4)) • 0—rssi < -110 dBm • 1—-110 dBm ≤ rssi < -109 dBm • 2—-109 dBm ≤ rssi < -108 dBm • ... • 61—-50 dBm ≤ rssi < -49 dBm • 62—-49 dBm ≤ rssi < -48 dBm • 63—-48 dBm ≤ rssi • 99—Not known or not detectable <ber> (Channel bit error rate, in percent) • 0–7—As RXQUAL values in the table in 3GPP TS 45.008 [20] subclause 8.2.4 • 99—Not known or not detectable <rscp> (Received signal code power (see 3GPP TS 25.133 [95] subclause 9.1.1.3 and 3GPP TS 25.123 [96] subclause 9.1.1.1.3)) • 0—rscp < -120 dBm • 1—-120 dBm ≤ rscp < -119 dBm • 2—-119 dBm ≤ rscp < -118 dBm • ... • 94—-27 dBm ≤ rscp < -26 dBm • 95—-26 dBm ≤ rscp < -25 dBm • 96—-25 dBm ≤ rscp • 255—Not known or not detectable (Continued on next page)

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+CESQ (continued)	**Extended signal quality (continued)** <ecno> (Ratio of the received energy per PN chip to the total received power spectral density (see 3GPP TS 25.133 [95] subclause)) • 0—Ec/Io < -24 dBm • 1—-24 dBm ≤ Ec/Io < -23.5 dBm • 2—-23.5 dBm ≤ Ec/Io < -23 dBm • ... • 47—-1 dBm ≤ Ec/Io < -0.5 dBm • 48—-0.5 dBm ≤ Ec/Io < 0 dBm • 49—0 dBm ≤ Ec/Io • 255—Not known or not detectable <rsrq> (Reference signal received quality (see 3GPP TS 36.133 [96] subclause 9.1.7)) • 0—rsrq < -19.5 dBm • 1—-19.5 dBm ≤ rsrq < -19 dBm • 2—-19 dBm ≤ rsrq < -18.5 dBm • ... • 32—-4 dBm ≤ rsrq < -3.5 dBm • 33—-3.5 dBm ≤ rsrq < -3 dBm • 34—-3 dBm ≤ rsrq • 255—Not known or not detectable <rsrp> (Reference signal received power (see 3GPP TS 36.133 [96] subclause 9.1.4)) • 0—rsrp < -140 dBm • 1—-140 dBm ≤ rsrp < -139 dBm • 2—-139 dBm ≤ rsrp < -18 dBm • ... • 95—-46 dBm ≤ rsrp < -45 dBm • 96—-45 dBm ≤ rsrp < -44 dBm • 97—-44 dBm ≤ rsrp • 255—Not known or not detectable

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+CGACT	**Activate/deactivate PDP context** **Notes:** • Up to three (3) PDP contexts can be active at once. • Any PDN which is not activated by CGACT AT command cannot be deactivated using the CGACT AT command. **Password required:** No **Usage:** • Execution: **AT+CGACT=[<state>[, <cid>[, <cid>[, ...]]]]** Response: OK *or* CME ERROR: <err> Purpose: Activate or deactivate the specified PDP contexts. • Query: **AT+CGACT?** Response: [+CGACT: <cid>, <state>] [+CGACT: <cid>, <state>]... OK Purpose: Display the activation states of all defined PDP contexts. • Query List: **AT+CGACT=?** Purpose: Return the supported <state> values. **Parameters:** <cid> (PDP context identifier) • Valid range: 1–24. • Maximum # of usable PDP contexts: 16 <state> (PDP context activation state) • 0—Deactivated • 1—Activated

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+CGAUTH	**Set/Report PDP connection authentication parameters** Set or report the authentication parameters for a PDP context. The context is identified by the supported profile that was used during the PDP context activation and PDP context modification procedures. **Password required:** No **Usage:** • Execution: **AT+CGAUTH=<cid>,<auth_prot>[, <userid>,<password>]** Response: OK *or* ERROR Purpose: Set the required authentication type and related values for the specified PDP profile (<cid>). • Query: **AT+CGAUTH?** Response: +CGAUTH: <cid>, <auth_prot>[,<userid>] ... OK Purpose: Display the authentication type and (if required) the username required for each profile. (Note: The <password> does not appear, for security reasons.) • Query List: **AT+CGAUTH=?** Purpose: Return the execution command format and the supported parameter values. **Parameters:** <cid> (PDP context identifier) • Valid range: 1–24. • Maximum # of usable PDP contexts: 16 <auth_prot> (Required authentication type) • 0—None. Username and password are not required. • 1—PAP. Username and password accepted • 2—CHAP. Username and password (secret) accepted <userid> (Username for PAP/CHAP authentication) • ASCII string within quotes (e.g. "userid") • Required for <auth_type> 1 (PAP) and 2 (CHAP) <password> (Password for PAP/CHAP authentication) • ASCII string within quotes (e.g. "123456") • Required for <auth_type> 1 (PAP) and 2 (CHAP)

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+CGDCONT	**Define PDP context** Define PDP (Packet Data Protocol) parameter values for a specific PDP context. **Notes:** • This implementation of +CGDCONT is derived from the 3GPP TS 27.007 version 13.2.0 specification, but does not support the full set of parameters from the specification and has extended usage rules. **Password required:** No **Usage:** • Execution: **AT+CGDCONT=<cid>[, <PDP_type> [, <apn> [, <PDP_addr> [, <d_comp> [, <h_comp> [, <pd1> [... [, <pdN>]]]]]]]]** Response: OK Purpose: Set the specified parameter values for the PDP context identified by <cid>. If only <cid> is specified, all parameter values are stored as undefined. • Query: **AT+CGDCONT?** Response: +CGDCONT: <cid>, <PDP_type>, <apn>, <PDP_addr>, <d_comp>, <h_comp>[, <pd1>[, ...[,<pdN>]]] ... OK Purpose: Report the current settings for each defined PDP context. • Query List: **AT+CGDCONT=?** Purpose: Return the execution command format and the supported parameter values. If multiple PDP types (<PDP_type>) are supported, the parameters for each <PDP_type> are returned on a separate line. **Parameters:** <cid> (PDP context identifier) • Valid range: 1–16 • Maximum # of usable PDP contexts: 16 <PDP_type> (Packet Data Protocol type) • "IP"—Internet Protocol, version 4 (IETF STD 5) • "IPV6"—Internet Protocol, version 6 (IETF RFC 2460) • "IPV4V6"—Virtual type that handles dual IP stack UE capability (3GPP TS 24.301[83]) Note: IPv4v6 is compliant up to 3GPP Release 7. <APN> (Access Point Name) • ASCII string within quotes • Logical name used to select GGSN or external packet data network • If null or omitted, subscription value will be requested (Continued on next page)

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+CGDCONT (continued)	**Define PDP context (continued)** <PDP_addr> (Access Point Name) • ASCII string within quotes • Identifies the MT in the address space applicable to the PDP. • If the value is null or omitted then a value may be provided by the TE during the PDP startup procedure or, failing that, a dynamic address will be requested. The READ command will continue to return the null string even if an address has been allocated during the PDP startup procedure. The allocated address may be read using the +CGPADDR command. • When +CGPIAF is supported, its settings can influence the format of this parameter returned with the read form of +CGDCONT. • Note: The value of this parameter is ignored with the set command. The parameter is included in the set command for backwards compatibility reasons only. <d_comp> (Data compression) • Applies to SNDCP (Sub Network Dependent Convergence Protocol) only • 0—(Default) Off. • 1—On (Manufacturer preferred compression) • 2—V.42 bis <h_comp> (PDP header compression) • 0—(Default) Off. • 1—On (Manufacturer preferred compression) • 2—RFC1144 (applies to SNDCP only) • 3—RFC2507 • 4—RFC3095 (applies to PDCP only) <pd1>, ... <pdN> (<PDP_type>-specific values)) • Zero to N string parameters • Parameter meanings are specific to <PDP_type>

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+CMEE	**Report mobile termination error** Select the method for reporting errors—+CME ERROR with result code, or ERROR. **Notes:** • Session must be closed using +KHTTPCLOSE before using this command. **Password required:** No **Usage:** • Execution: **AT+CMEE=[<n>]** Response: OK Purpose: Select the error reporting method. • Query: **AT+CMEE?** Response: +CMEE: <n> OK Purpose: Display the current error reporting method. • Query List: **AT+CMEE=?** Purpose: Display valid execution format and parameter values. **Parameters:** <n> (CME error reporting state) • 0—Disabled. Use "ERROR." • 1—Enabled. Use "+CME ERROR: <err>" with numeric <err> result codes.

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+CPSMS	**Configure Power Saving Mode (PSM)** Enable/disable and configure the UE's Power Saving Mode parameters. **Notes:** • This implementation of +CPSMS follows 3GPP TS 27.007, with exceptions as noted in the parameter descriptions. **Password required:** No **Usage:** • Execution: **AT+CPSMS=<mode>[, <Requested_Periodic-RAU>], [<Requested_GPRS-READY-timer>], [<Requested_Periodic-TAU>], [<Requested_Active-Time>]** Response: OK *or* +CME ERROR: <err> Purpose: Enable/disable PSM, and configure PSM settings. • Query: **AT+CPSMS?** Response: +CPSMS: <mode>, [<Requested_Periodic-RAU>], [<Requested_GPRS-READY-timer>], [<Requested_Periodic-TAU>], [<Requested_Active-Time>] OK Purpose: Report current PSM status and settings. • Query List: **AT+CPSMS=?** Purpose: Return the execution command format and the supported parameter values. **Parameters:** <mode> (Enable/Disable PSM) • 0—Disable PSM • 1—Enable PSM <Requested_Periodic-RAU> (3G Routing Area Update timer) • Leave blank, not used. <Requested_GPRS-READY-timer> (2G timer) • Leave blank, not used. <Requested_Periodic-TAU> (TAU timer—Amount of time UE will be dormant before timer wakes it) • One byte (8 bits) represented as a string. For coding and value range details, refer to the +CPSMS description in 3GPP TS 27.007. • Default—"00011000"=4 hours • e.g. "01000111" = 70 hours <Requested_Active-Time> (Amount of time UE will remain active (idle) before re-entering PSM) • One byte (8 bits) represented as a string. For coding and value range details, refer to the +CPSMS description in 3GPP TS 27.007. • Default—"00001010"=20 seconds • e.g. "00100100" = 4 minutes

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+CPWROFF	**Power Off** **Notes:** • If no <mode> is specified for the execution command, the module sends an IMSI detach request to the network before powering down. • If <mode>=1 is specified for the execution command, the module performs a fast power down (~1s faster than not specifying the <mode>) without sending an IMSI detach request to the network. • The module can be woken by setting POWER_ON_N low to turn on the system. **Password required:** No **Usage:** • Execution: **AT+CPWROFF [=<mode>]** Response: OK *or* ERROR Purpose: Power off the module using selected mode. • Query List: **AT+CPWROFF=?** Purpose: Display currently selected power off mode. **Parameters:** <mode> (Power off mode) • 1—Fast power down mode

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+CSQ	**Display signal quality** Display the current signal strength and BER. **Password required:** No **Usage:** • Execution: **AT+CSQ** Response: +CSQ: <rssi>,<ber> OK *or* +CME ERROR: <err> Purpose: Display the current signal strength and BER. • Query List: **AT+CSQ=?** Purpose: Display the range of possible values for signal strength and BER. **Parameters:** <rssi> (Received Signal Strength Indication offset value) • Integer value. Each step represents 2 dBm increase from base value • 0: -113 dBm or less • 1–30: -111 to -53 dBm • 31: -51 dBm or greater • 99: Not known, or not detectable <ber> (Channel Bit Error Rate, in percent) • Integer value. • 0–7: As RXQAL values in the table in 3GPP TS 45.008 subclause 8.2.4 • 99: Not known, or not detectable

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!CUSTOM	**Set/return customization settings** Set or return several customization values. **Notes:** • Some customizations may not be available for certain chipsets, firmware revisions, or devices. **Password required:** Yes (Execution only) (see !ENTERCND for details) **Usage:** • Execution: **AT!CUSTOM=<customization>, <value>** Response: OK Purpose: Display the customization and value. • Query: **AT!CUSTOM?** Response: (list of enabled <feature>s) OK Purpose: Display features that are currently enabled. • Query list: **AT!CUSTOM=?** Response: (list of all <customization>s supported) !CUSTOM: "WAKEHOSTEN" "USBSERIALENABLE" "LOGENABLE" "CFUNPERSISTEN" "UIM2ENABLE" "SIMPOWERSAVE" OK Purpose: Return a list of valid <customization> values. **Parameters:** <value> (Value being assigned to a specific <customization> setting) • Value of customization as decimal value. For hex input, precede value with '0x' <customization> (String identifying customization setting. The default value for all customizations is 0.) *Note:* • *Use quotation marks around the customization string. For example, AT!CUSTOM="CSDOFF",0.* • *Some commands are for SWI internal use only.* • "CFUNPERSISTEN"—Enable/disable persistence (across power cycles) of AT+CFUN setting. • <value>: • 0—Disable (+CFUN setting does not persist across power cycle) Note: If the modem is in P-LPM (persistent low power mode—AT+CFUN mode 0) when this option is used, persistence remains enabled until the modem is put into online mode using an AT or QMI command. • 1—Enable (+CFUN setting persists across power cycle) • Note: This customization does not affect operating mode persistence set using other interfaces. For example, the QMI interface can still be used to set the operating mode to LPM or P-LPM, even if this customization is disabled. (Continued on next page)

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!CUSTOM (continued)	**Set—query customization settings (continued)** • "LOGENABLE"—Disable/enable logs. <value>: • 0x00—Disable • 0x01—Enable • Note: When you set LOGENABLE, it only becomes effective on the next boot. For 0x00, device reset is required for LOG serial port reconnection. • "SIMPOWERSAVE"—Enable/disable SIM power save. <value>: • 0x00—Disable (Default) • 0x01—Enable *Note: SIMPOWERSAVE must be enabled to enable the SIM to enter PSM when the module is in sleep state. If SIMPOWERSAVE is disabled, the module will not be able to enter Lite-Hibernate or Hibernate. For power mode control details, see +KSLEEP.* • "UIM2ENABLE"—Enable/disable UIM2 slot support. <value>: • 0—Disable (Default) • 1—Enable • "USBSERIALENABLE"—Use serial number in USB Descriptor <value>: • 0—Default (same as 1) • 1—Use IMEI as serial number in USBD • 2—Do not use any serial number in USBD • "WAKEHOSTEN"—Enable/disable host wake-up via SMS. <value>: • 0—Disabled • 1—Enabled *Note: Detecting TCP / UDP through the WAKE_ON_WWAN pin is not supported.*

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!GSTATUS	**Return operational status** Return specific details about the current operational status of the modem. **Important:** *Response details may vary from release to release. Parameter descriptions show all possible values. Contact Sierra Wireless for further details if required.* **Password required:** No **Usage:** • Query: **AT!GSTATUS?** Response (As noted above, details may vary): !GSTATUS: <param_label>: <param> [[<param_label>:]<param>] ... OK Purpose: Display details about the modem's current operational state. Details shown will vary depending on the module type and firmware release. **Example:** !GSTATUS: Current Time: <ctime> Temperature: <temp> Reset Counter: <rstcount> CFUN Mode: <mode> System mode: <smode> PS state: <PSstate> LTE Band: <band> LTE Rx ARFCN:<dlchannel> LTE BW: <bandwidth> LTE Tx ARFCN:<ulchannel> PCI: <pci> RSSI: <rssi> RSRP: <rsrp> RSRQ: <rsrq> SINR: <sinr> Tx Power: <tx power> EMM state: <emm state> EMM Mode: <emm mode> RRC State: <rrc_state> TAC: <tac> Cell ID: <cid> OK **Parameters:** <param_label> • Parameter description. e.g. "LTE Band" <param> • Parameter value. Refer to the parameter descriptions listed below. (Continued on next page)

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!GSTATUS (continued)	**Return operational status (continued)** <ctime> ("Current time") • ASCII string • Format: "yyyy/MM/dd,hh:mm:ss±zz", where: • yyyy—Year (e.g. 2023) • MM—Month (01–12) • dd—Day (01–31) • hh—Hour (00–23) • mm—Minute (00–59) • ss—Second (00–59) • zz—Time zone (-96~+96) representing the number of quarter-hours (15 minute blocks) between local time and GMT. • e.g., "2023/09/06,22:10:00+08" = September 06, 2023, 22:10:00 GMT + 2 hours (which is September 07, 2023, 00:10:00 local time) <temp> ("Temperature"—Approximate temperature in °C, accurate within ~5 °C) • 32-bit decimal <rstcount> ("Reset Counter"—Number of resets since the last power cycle) • 32-bit decimal • Value resets to 0 when the module power cycles (powers off and on). • Value increases by 1 when a software reset is performed. <Mode> ("CFUN Mode"—Current +CFUN mode) • 0—Minimum functionality • 1—Full functionality • 4—Turn off RF <smode> ("System Mode"—current RAT) • ASCII string. Valid values: • "LTE" • "None" <PSstate> ("PS State"—PS attached state) • 0—Detached • 1—Attached <band> ("LTE Band"—Current LTE band being processed) • Decimal • 3GPP band number (See Table 13-1 for details.) <dlchannel> ("LTE Rx ARFCN") • Decimal <lbw> ("LTE BW"—LTE bandwidth) • ASCII string (quotation marks do not appear): • "1.4 MHz" • "3 MHz" • "5 MHz" • "10 MHz" • "15 MHz" • "20 MHz" <ulchannel> ("LTE Tx ARFCN") • Decimal (Continued on next page)

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!GSTATUS (continued)	**Return operational status (continued)** <pci> ("PCI"—Physical Cell Identifier) • Decimal value • Valid range: 0–503 <rssi> ("RSSI"—Total received power) • -120 to 0 <rsrp> ("RSRP"—Reference Signal Receive Power, in dBm) • -140 to -44 <rsrq> ("RSRQ"—Reference Signal Receive Quality, in dB) • -20 to -3 <sinr> ("SINR"—Signal to Interference plus Noise, in dB) • -20 to +30 <tx power> ("Tx Power") • Decimal <emm state> ("EMM state"—Current EMM state) • "NULL" • "DEREG" • "REG INIT" • "REG" • "DEREG INIT" • "TAU INIT" • "SR INIT" • "UNKNOWN" <emm mode> ("EMM Mode" first field—Current EMM state) • "IDLE" • "PSM" • "CONNECTED" <rrc_state> (Radio Resource Control (RRC) state) • "DEACT" • "OOS" • "IDLE" • "SUSPEND IDLE" • "CONNECTED" • "UNKNOWN" <tac> ("TAC"—Tracking Area Code) • Decimal <cid> ("Cell ID"—Cell ID) • Decimal

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!HWID	**Read hardware ID** Return the module's hardware ID, which combines the major and minor version number. **Password required:** Yes **Usage:** • Query: **AT!HWID?** Response: Revision: <MajorVer>.<MinorVer> OK Purpose: Display the module's hardware ID. • Query List: **AT!HWID=?** Purpose: Display valid execution format and parameter values. **Parameters:** <MajorVer> (Major version number) • Valid range: 0–9 <MinorVer> (Minor version number) • Valid range: 0–9 **Example:** • AT!HWID? Revision: 1.7 OK

Table 3-2: Modem Status Command Details (Continued)

Command	Description
I	**Display product identification information** Display the module's hardware and firmware identification information. **Password required:** No **Usage:** • Query: **ATI** Response Manufacturer: <manufacturer> Model: <model> Revision: <revision> SVN: <svn> IMEI: <imei> FSN: <fsn> TS.25: <ts25 version> +GCAP: <gcap> OK Purpose: Display the module's hardware or firmware information. **Parameters:** <manufacturer> (See AT+GMI) • ASCII string <model> (See AT+GMM) • ASCII string • Integer <revision> (See AT+GMR) • ASCII string <svn> (Software Version Number) • Hex string <imei> (Mobile Equipment Identifier) • Hex string <fsn> (Factory Serial Number) • ASCII string <ts25 version> (TS.25 table version) • ASCII string • Note: TS.25 is a table maintained by the GSMA that is used for MCC/MNC and operator name mapping. <gcap> (Device Capabilities List) • ASCII string

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+KCELL	**Display Detected Cell Details** Display information about the LTE serving cells detected by the module. **Password required:** No **Usage:** • Execution: **AT+KCELL=<revision>** Response: +KCELL: <nbLTEcells>[,<cell_type>,<Earfcn>,<PLMN>,<LTE_CI>, <PhyCellInd>,<trackingAreaCode>,<RSRPResult>,<RSRQResult>] [,<cell_type>,<Earfcn>,<PLMN>,<LTE_CI>,<PhyCellInd>, <trackingAreaCode>,<RSRPResult>,<RSRQResult>][…] OK Purpose: Display details about all LTE serving cells detected by the module. • Query: **AT+KCELL?** Response:! OK Purpose: Display execution format. • Query list: **AT+KCELL=?** Purpose: Displays execution format. **Parameters:** <revision> (Reserved field) • 0—Only valid option. Parameter is reserved for future development. <cell_type> (Cell type) • 5—LTE serving cell <PLMN> (PLMN identifier) • Format: Hexadecimal (3 bytes) per GSM 11.11 specification • Combines MCC (Mobile Country Code) and MNC (Mobile Network Code) • Example: 42F618 (Hex value for MCC=246 and MNC=81) <nbLTEcells> (Number of available LTE base stations) • Valid range: 0–33 <LTE_CI> (LTE Cell Identity) • Format: Hexadecimal (8 hex digits; length 28 bits), per 3GPP TS 36.331, 6.3.4, Cell Identity • Example: A12BC3DF <TrackingAreaCode> (Tracking Area Code of LTE Cell) • Valid range: 0–65535, per 3GPP TS 36.331, 6.3.4, TrackingAreaCode <RSRPResult> (Reference Signal Received Power) • Valid range: 0–97. Refer to 3GPP TS 36.331, 6.3.5, RSRP-Range for details. <RSRQResult> (Reference Signal Received Quality) • Valid range: 0–34. Refer to 3GPP TS 36.331, 6.3.5, RSRQ-Range for details. <Earfcn> (Neighbor cell carrier frequency • Carrier frequency of the neighbor cell designated by the EUTRA Absolute Radio Frequency Channel Number (EARFCN). Refer to 3GPP TS 36.101, 5.7.3 for details. • Valid range: 0–0xFFFF <PhyCellInd> (Physical Cell ID) • Valid range: 0–503, per 3GPP TS 36.331, 6.3.4, PhysCellId IE

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+KGSN	**Request Product Serial Number Identification and Software Version** **Notes:** • This command has been developed to provide the IMEI SV and Serial Number through an AT command and it can work without a SIM. **Password required:** No **Usage:** • Execution : **AT+KGSN=<number_type>** Response (<number_type>=0): +KGSN: <IMEI> OK Response (<number_type>=1): +KGSN: <IMEISV> OK Response (<number_type>=2): +KGSN: <IMEISV_STR> OK Response (<number_type>=3): +KGSN: <FSN> OK Response (<number_type>=4): +KGSN: <FSN-BB> OK Purpose: Display the requested information type. • Query List: **AT+KGSN=?** Purpose: Display valid parameter values. **Parameters:** <number_type> (Information type to display) • valid range: 0–4 <IMEI> (15-digit IMEI) • Format: <8-digit TAC> + <6-digit SNR> + <1-digit check> • e.g. 351578000023006 <IMEISV> (16-digit IMEISV) • Format: <8-digit TAC> + <6-digit SNR> + <2-digit SVN> • e.g. 3515780000230001 <IMEISV_STR> (Formatted IMEISV string) • Format: <15-digit + 1 check digit> SV: <software version> • e.g. 35157800002300-6 SV:01 <FSN> (14-character serial number) • String • e.g. 0123456789ABCD <FSN-BB> (14-character serial number) + 2-digit batch revision) • String • e.g. 0123456789ABCD01

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+KNTPCFG	**Configure SNTP Client** Configure the SNTP CLIENT. **Password required:** No **Persistent across power cycles:** Yes **Usage:** • Execution: **AT+KNTPCFG=[<mode>],[[<NTPUrl>],<updateClk>],[<Timezone>]** Response: OK Purpose: Configure the SNTP client. • Query: **AT+KNTPCFG?** Response: +KNTPCFG: <mode>,<NTPUrl>,<updateClk>,<Timezone> OK Purpose: Display the current SNTP client configuration. • Query list: **AT+KNTPCFG=?** Purpose: Return the execution command format and supported parameter values. **Parameters:** <mode> (SNTP client mode) • 0—Disable (Default) • 1—Enable <NTPUrl> (NTP server URL) • The NTP client queries the time from this URL. • ASCII string; maximum length 255 characters • If not configured (using the Execution format), the Query response will return a null string. • Format: "<host_url>[:<port>]" (quotation marks are not required but can be used) where: • <host_url>: FQDN (e.g., "www.google.com") • <port>: 1–65535 (Default: 80) • e.g., "www.google.com", "www.google.com:80" <updateClk> (Update the device clock with the NTP time) • 0—Disable (Default) • 1—Enable <Timezone> (Difference between local time and UTC, in quarters of an hour) • Valid range: -48 to +56 • Default: 0

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+KSLEEP	**Configure UART1 power management (sleep mode entry conditions)** Configure UART1 power management, indicating under which conditions the module will enter sleep mode. **Password required:** No **Persistent across power cycles:** Yes **Notes:** • Controls only UART1 power management; does not affect USB AT command port. • When KSLEEP=1 and the module is in sleep mode, the user must input a character to wake the module. When the module is awake, AT commands can be input as normal. **Usage:** • Execution: **AT+KSLEEP=<mngt>[,<level>[,<delay>]]** Response: OK Purpose: Set the power management configuration. • Query: **AT+KSLEEP?** Response: +KSLEEP: <mngt>,<level>,<delay> OK Purpose: Indicate current power management configuration. • Query list: **AT+KSLEEP=?** Purpose: Return the execution command format and supported parameter values. **Parameters:** <mngt> (UART1 Power management configuration) • 0—Module will not enter sleep mode when DTR is active (low level). If DTR is inactive, module enters sleep mode once all wakeup sources are released. Note: DTR must be active to send AT commands. • 1—Module enters sleep mode automatically after 5 seconds of inactivity. • 2—Module never enters sleep mode (regardless of DTR state) <level> (Lowest power saving mode that the module can enter) • Mandatory when <mngt>=0 or 1; not allowed when <mngt>=2 • 0—Sleep • 1—Lite Hibernate • 2—Hibernate • Note—The actual power saving level applied to the module may be less than the configured level, due to other subsystem requirements and hardware limitations. <delay> (Length of delay before the module enters power saving mode after reboot, in ms) • Valid range: 5000—65535 *Note: The !CUSTOM option "SIMPOWERSAVE" must be enabled (set to to enable the SIM to enter PSM when the module is in sleep state. If SIMPOWERSAVE is disabled, the module will not be able to enter Lite-Hibernate or Hibernate. For power mode control details, see +KSLEEP.* *Enabling SIMPOWERSAVE lowers the current draw in sleep mode.*

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+KSREP	**Enable/disable startup reporting** Enable or disable startup reporting. When enabled, the module sends an unsolicited notification (+KSUP (notification)) during startup. By default, startup reporting is disabled. **Password required:** No **Usage:** • Execution: **AT+KSREP=<mode>** Response: OK Purpose: Enable or disable startup reporting. • Query: **AT+KSREP?** Response: +KSREP: <mode>,<status> OK Purpose: Report current setting for startup reporting, and the current status. • Query List: **AT+KSREP=?** Response: +KSREP: (list of supported <mode>s) OK Purpose: Return the execution command format. See the parameter descriptions below for details. **Parameters:** <mode> (Startup reporting state) • 0 (Default)—Disabled • 1—Enabled <status> (Module status) • 0—Module is ready to receive commands for the TE. No access code is required. • 1—Module is waiting for an access code. Use AT+CPIN? to determine the code. • 2—SIM card is not present. • 3—Reserved • 4—Unrecoverable error • 5—Unknown state

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+KSUP (notification)	**Startup notification (unsolicited notification)** Unsolicited notification received from the module at startup, if enabled using +KSREP. **Usage:** • Notification: **+KSUP: <status>** Purpose: Indicates the state of the module at startup time. **Parameters:** <status> (Module status) • 0—Module is ready to receive commands for the TE. No access code is required. • 1—Module is waiting for an access code. Use AT+CPIN? to determine the code. • 2—SIM card is not present. • 3—Reserved • 4—Unrecoverable error • 5—Unknown state

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!NETDEVCTL	**Bind/Unbind IP address with host for NIC USB interface** Bind/unbind a data connection to a PDP context. **Password required:** No **Usage:** • Execution: **AT!NETDEVCTL=<op>,<cid>[,<urc_en>]** Response: OK Purpose: Bind or unbind the connection for the specified <cid>. • Query: **AT!NETDEVCTL?** Response: !NETDEVCTL: <op><cid>,<urc_en>,<status> OK Purpose: Display current control setting. • Query list: **AT!NETDEVCTL=?** Purpose: Display valid execution format and parameter values. **Parameters:** <op> (Device control option) • 0—Unbind <cid> for host NIC interface • 1—Bind <cid> for host NIC interface once and no rebind if reactivate PDN context with the same <cid> after deactivation. • 2—Bind <cid> for host NIC interface and rebind if reactivate PDN context with the same <cid> after deactivation. • 3—Auto-dial and bind <cid> when power on, saved in NV after power down. <cid> (Context ID) • Valid range: 0–15 <urc_en> (URC state) • 0—Disable URC • 1—Enable URC <status> (Binding status) • 0—Bind <cid> for host NIC interface failure • 1—Bind <cid> for host NIC interface success **Example(s):** AT!USBCOMP=1,2,1019 OK AT!RESET OK AT!NETDEVCTL=3,1,1 OK NETDEVCTL: 1

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!NVBACKUP	**Backup NV data** **Password required:** Yes (for <category=2>) **Usage:** • Execution: **AT!NVBACKUP=<category>]** Response: !NVBACKUP: NV Items Saved: <saved> [NV Items Skipped: <skipped>] OK Purpose: Perform the specified backup type. **Parameters:** <category> (Backup type) • 2—OEM/User • 3—Cache (Boot and frequently-updated NV items) <saved> (Number of NV items saved) • Range: 0–255 <skipped> (Number of NV items skipped) • Range: 1–255 • The 'skipped' response line does not appear if there were no items skipped.

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!PATEMP	**Return PA temperature information** Return the module's PA temperature state and current temperature. **Password required:** No **Usage:** • Query: **AT!PATEMP?** Response: Temp state: <state> Temperature: <temperature> degC OK Purpose: Return the module's Power control temperature information. **Parameters:** <state> (Temperature state): • Valid values: ◦ "Normal" ◦ "Low Warning" ◦ "High Warning" ◦ "Low Critical" ◦ "High Critical" <temperature> (Current temperature): • Decimal ASCII string • Current PA temperature in degrees Celsius. This is the temperature reported by a thermistor positioned near the power amplifiers. • Example: "32.3"

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!PATEMP (notification)	**PA temperature state change—Unsolicited notification** Unsolicited notification received when the PA temperature state changes. To enable !PATEMP (and other notifications), use AT+WUSLMSK. **Notification format:** **!PATEMP: <state>** **Parameters:** <state> (PMIC temperature state) • Valid range: 1–5 • 1—Normal • 2—Low Warning • 3—High Warning • 4—Low Critical • 5—High Critical

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!PCINFO	**Return power control status information** Return the modem's power control status information. **Password required:** No **Usage:** • Query: **AT!PCINFO?** Response: State: <state> LPM voters – Temp:<lpmvote>, Volt:<lpmvote>, User: <lpmvote>, W_DISABLE: <lpmvote>, LWM2M: <lpmvote> LPM persistence - (list of enabled <lpmname>s:<lpmpersist>) OK Purpose: Return power control information. **Parameters:** <state> (The modem's power mode) • ASCII string (quotation marks do not appear): • "Initialization" • "Low Power Mode" • "LPM in Progress" • "Online" • "Online In Progress" <lpmname> (Client name) • Temp: PC Temp client • Volt: PC Volt client • User: User client - AT • W_DISABLE: W_DISABLE client • LWM2M: Lightweight M2M <lpmvote> (Client's LPM vote) • Range: 0—1 • 0: No vote • 1: Vote <lpmpersist> (Client's LPM persistence) • 1: Current state of user-initiated Low Power Mode **Examples:** • If all clients do not have the persistence, set: AT!PCINFO? State: Online LPM voters - Temp:0, Volt:0, User:0, W_DISABLE:0, LWM2M:0 LPM persistence - None OK • If one or more clients do have the persistence, set: AT!PCINFO? State: Low Power Mode LPM voters - Temp:1, Volt:0, User:0, W_DISABLE:1, LWM2M:0 LPM persistence - W_DISABLE: 1 OK

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!PCTEMP	**Return Power control temperature information** Return the module's power control temperature state and current temperature. **Password required:** No **Usage:** • Query: **AT!PCTEMP?** Response: Temp state: <state> Temperature: <temperature> degC OK Purpose: Return the module's power control temperature information. **Parameters:** <state> (Temperature state): • Valid values: ◦ "Normal" ◦ "Low Warning" ◦ "High Warning" ◦ "Low Critical" ◦ "High Critical" <temperature> (Current temperature): • Decimal ASCII string • Current temperature in degrees Celsius. • Example: "32.3"

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!PCTEMP (notification)	**PMIC temperature state change—Unsolicited notification** Unsolicited notification received when the PMIC temperature state changes. To enable !PCTEMP (and other notifications), use AT+WUSLMSK. **Notification format:** **!PCTEMP: <state>** **Parameters:** <state> (PMIC temperature state) • Valid range: 1–5 • 1—Normal • 2—Low Warning • 3—High Warning • 4—Low Critical • 5—High Critical

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!PCTEMPLIMITS	**Set/report temperature state limit values** Certain modem functionality is affected by the modem's temperature state. The possible temperature states are high critical, high warning, high normal, low normal, and low critical. Use this command to report or set the limits that correspond to these temperature states. To display the current temperature and temperature state, see !PCTEMP on page 50. **Notes:** • All temperatures are in Celsius. **Password required:** Yes **Usage:** • Execution: **AT!PCTEMPLIMITS=<hc>,<hw>,<hn>,<ln>,<lc>** Response: OK Purpose: Set the temperature limits for each state (all five values must be specified). • Query: **AT!PCTEMPLIMITS?** Response: HI CRIT: <hc> HI WARN: <hw> HI NORM: <hn> LO NORM: <ln> LO CRIT: <lc> Purpose: Return the temperature limits for each state. **Parameters:** <hc> (High Critical) • Temperature limit varies by device (refer to the RC71xx Product Technical Specification). <hw> (High Warning) • Temperature limit varies by device (refer to the RC71xx Product Technical Specification). <hn>(High Normal) • Temperature limit varies by device (refer to the RC71xx Product Technical Specification). <ln> (Low Normal) • Temperature limit varies by device (refer to the RC71xx Product Technical Specification). <lc> (Low Critical) • Temperature limit varies by device (refer to the RC71xx Product Technical Specification).

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!PCVOLT	**Return current power supply voltage information** Return the module's power control supply state and actual voltage. **Password required:** No **Usage:** • Query: **AT!PCVOLT?** Response: Volt state: Normal Power supply voltage: <voltage> mV (<raw> cnt) OK Purpose: Return the module's voltage information. **Parameters:** <state> (Power supply state): • Valid values: ◦ "Normal" ◦ "Low Warning" ◦ "High Warning" ◦ "Low Critical" ◦ "High Critical" <voltage>: • Current voltage reading in mV. • Decimal ASCII <raw>: • ADC (Analog/digital convertor) reading • Decimal ASCII

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!PCVOLT (notification)	**PMIC voltage state change—Unsolicited notification** Unsolicited notification received when the PMIC voltage state changes. To enable !PCVOLT (and other notifications), use AT+WUSLMSK. **Notification format:** **!PCVOLT: <state>** **Parameters:** <state> (Power supply state) • Valid range: 1–5 • 1—Normal • 2—Low Warning • 3—High Warning • 4—Low Critical • 5—High Critical

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!PCVOLTLIMITS	**Set/report power supply voltage state limit values** Certain modem functionality is affected by the modem's power supply voltage state. The possible voltage states are high critical, high normal, low normal, low warning, and low critical. Use this command to report or set the limits that correspond to these voltage states. **Password required:** Yes **Usage:** • Execution: **AT!PCVOLTLIMITS=<hc>,<hn>,<ln>,<lw>,<lc>** Response: OK Purpose: Set the voltage limits for each state (all five values must be specified). • Query: **AT!PCVOLTLIMITS?** Response: HI CRIT: <hc> HI NORM: <hn> LO NORM: <ln> LO WARN: <lw> LO CRIT: <lc> Purpose: Return the voltage limits for each state. **Parameters:** <hc> (High Critical) • Voltage limit varies by device (refer to the RC71xx Product Technical Specification) <hw> (High Normal) • Voltage limit varies by device (refer to the RC71xx Product Technical Specification) <ln> (Low Normal) • Voltage limit varies by device (refer to the RC71xx Product Technical Specification) <lw> (Low Warning) • Voltage limit varies by device (refer to the RC71xx Product Technical Specification) <lc> (Low Critical) • Voltage limit varies by device (refer to the RC71xx Product Technical Specification)

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!POWERDOWN	**Power down system** Power down the system. **Password required:** No **Usage:** • Execution: **AT!POWERDOWN** Response: OK Purpose: Power the system down.

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!POWERMODE	**Enable/disable PSM** Enable or disable power saving mode (PSM). When this command is used to enable PSM, the +CPSMS parameters will be renegotiated with the network at the same time. A network-connected device will enter PSM only if the PSM parameter's negotiation succeeds. (The parameters (timers) specified in +CPSMS are requested values—the PSM negotiation determines the actual timer values that will be used.) **Password required:** No **Requirements:** • AT!POWERWAKE must be used to configure wakeup sources before using this command to enable PSM. **Usage:** • Execution: **AT!POWERMODE=<mode>** Response: OK Purpose: Enable/disable PSM. • Query: **AT!POWERMODE?** Response: !POWERMODE: <mode> Purpose: Display the current state of PSM. • Query List: **AT!POWERMODE=?** Purpose: Return the execution command format and the supported parameter values. **Parameters:** <mode> (Power saving mode) • 0—Disable PSM. • 1—Enable PSM. When enabled, the module enters PSM, then begins monitoring for wakeup sources that were previously configured using !POWERWAKE. • To power down the module use !POWERDOWN.

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!POWERWAKE	**Configure PSM wakeup sources** Configure the wakeup sources (triggers) for Power Saving Mode (PSM)). When a module is in PSM, it is in a network-aware state. The module's low state is registered on the network and sleep time is negotiated. After configuring wakeup triggers, the command AT!POWERMODE can be used to enter PSM. **Notes:** • Timer must be configured for PSM mode. • At least one wakeup source must be configured before !POWERMODE can be used to select a power saving mode option that requires wakeup sources. • The PSM timer is not cleared by the "Execution (clear)" command format. **Password required:** No **Persistent across power cycles:** Partial (PSM timers persist) **Usage:** • Execution (timer): **AT!POWERWAKE=<type>,<psm>,[<active>]** Response: OK Purpose: Set the timeout period for a wakeup timer. • Query: **AT!POWERWAKE?** Response: !POWERWAKE:<type>,<psm>,<active> OK Purpose: Show currently configured wakeup sources. If a source is not configured, it will not appear. • Query List: **AT!POWERWAKE=?** Purpose: Return the execution command format and the supported parameter values. **Parameters:** <type> (Wakeup source type) • 1—Timer <psm> (Requested timer duration for staying in PSM) • Timer is the requested extended periodic TAU value (refer to +CPSMS). • Timer value must be greater than threshold specified in PSM configuration. • Max value: 35712000 (Timer duration in seconds) • Note: Power consumption may be impacted if a short timeout is used. <active> (Requested active timer duration, in seconds) • 0–11160—Active timer duration. The value indicates the period during which the device remains reachable for mobile-terminated (MT) transactions on transition from connected mode to idle mode. • If no value is specified, active time of 0 is configured. • Value must conform to GPRS Timer 2 IE in 3GPP TS 24.008.

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!PRIID	**Report module PRI part number and revision** Report the module's customer and carrier PRI part numbers and revisions. **Password required:** No **Usage:** • Query: **AT!PRIID?** Response: PRI Part Number: <priPn> Revision: <priRevDisplay> Customer: <priCustomer> Carrier PRI: <priFile> OK Purpose: Return the module's PRI part number (<priPn>) and revision (<priRevDisplay>). (In the example shown above, no Carrier PRI is present. If it were, then the Part Number and Revision would display.) **Parameters:** <priPn> (PRI part number) • 7-digit ASCII number • Example: 9991234 <priCustomer> (Customer name) • ASCII string • e.g. "Generic" (quotation marks do not appear) <priRevDisplay> (PRI revision number being read from the module) • ASCII string • Example: 000.000_000 <priFile> (PRI filename) • ASCII string • e.g. "SWI216C_01.02.01.03_GENERIC_000.001_000" (quotation marks do not appear)

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!RESET	**Reset modem** Perform a modem reset. **Password required:** No **Usage:** • Execution: **AT!RESET** Response: OK Purpose: Reset the modem.

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!SCUMMTU	**Report MTU Size** Report the MTU (maximum transmission unit) size. **Password required:** Yes **Notes:** • For AT&T, the MTU size is based on the SIM, per CDR-CDS-110 and CDR-CDS-112. **Usage:** • Query: **AT!SCUMMTU?** Response: !SCUMMTU: MTU : <mtu> OK Purpose: Display the current MTU size. **Parameters:** <mtu> (Maximum Transmission Unit, in bytes) • 0—Use default value • 576–2000—Other values required by carriers.

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!SKU	**Read Module SKU** Read the module's SKU value. **Password required:** Yes **Usage:** • Query: **AT!SKU?** Response: <value$_1$>,<value$_2$>,...,<value$_{40}$> OK Purpose: Display the test history 40-byte array. **Parameters:** <value> (Test history value) • Hex ASCII format • Range: 00–FF

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!USBCOMP	**Set/report USB interface configuration** Use this command with modems that have been configured with multiple USB compositions. By default, devices are typically configured to use a USB composition that presents a minimal set of interfaces to reduce end-user modem enumeration time. If the device also supports other compositions, this command can be used to build and select custom compositions from the supported interfaces. **Notes:** • Interface support may vary by product and firmware version. Use the Query List command to determine actual support. **Password required:** Yes (see !ENTERCND for details) **Usage:** • Execution: **AT!USBCOMP=<Config Index>,<Config Type>,<Interface bitmask>** Response: OK Purpose: Set the current composition. For the change to take effect, you must reset the modem. • Query: **AT!USBCOMP?** Response: Config Index: <Config Index> Config Type: <Config Type> (type string) Interface bitmask: <Interface bitmask> (interface string) OK Purpose: Report the current interface composition. • Query List: **AT!USBCOMP=?** Purpose: Display valid execution format and parameter values. <Config Index> (Configuration index to which composition applies) • Valid value(s): 1 <Config Type> (Configuration type) • Valid value(s): • 1—Generic • 2—Network Interface Card (NIC)—RNDIS or ECM • Note—Config Type 1 (Generic) supports only the serial interfaces (Log, AT, AT_PPP). Config Type 2 (Network Interface Card) supports the serial interfaces and NICs (RNDIS, ECM). <Interface bitmask> (Interfaces enabled for selected configuration) • Format: 32-bit bitmask • Valid values: • 00000001—Log • 00000008—AT. This interface cannot be disabled. (The command will return ERROR if this is not selected.) • 00000010—AT_PPP • 00001000—RNDIS • 00002000—ECM • Note—RNDIS and ECM are mutually exclusive. The bitmask cannot set both RNDIS and ECM.

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!USBINFO	**Return information from active USB descriptor** Return information from the active USB descriptor. **Password required:** No **Usage:** • Query: **AT!USBINFO?** Response: VID: <vendor_id> PID: <product_id> Manufacturer: <manuString> Product: <prodString> Purpose: Display USB descriptor information. **Parameters:** <vendor_id> (Vendor ID): • Valid range: 0000–FFFF <product_id> (Product ID used when modem is in application mode): • Valid range: 0000–FFFF <manuString> (Manufacturer string): • ASCII string (32 characters maximum) • Example: "Sierra Wireless, Incorporated" <prodString> (Product string): • ASCII string (64 characters maximum) • Examples: "RC7110", "RC7120"

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!USBPID	**Set/report product ID in USB descriptor** Use this command to set the device's product ID in the USB descriptor. (Some devices may support more than one product ID.) **Password required:** Yes (see !ENTERCND for details) **Reset required to apply changes:** Yes **Usage:** • Execution: **AT!USBPID=<product_id>** Response: OK Purpose: Set the product ID in the USB descriptor. • Query: **AT!USBPID?** Response: !USBPID: <Config Type>, <product_id>] OK Purpose: Report the product ID that is stored in the USB descriptor. • Query List: **AT!USBPID=?** Purpose: Display a list of default (non-custom) product IDs for the device. **Parameters:** <Config Type> • "GENERIC" • "NIC" (Network Interface Card) <product_id> • Hexadecimal ASCII value. • Valid range: 0000–FFFF **Example(s):** AT!USBPID=? GENERIC NIC 9104 9114 OK AT!USBPID=9104 OK AT!USBPID? !USBPID: GENERIC : 9104 OK AT!RESET OK

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!USBSHUTDOWN	**Force USB PHY Shutdown** To further reduce power consumption while the module is in Low Power Mode (Lite Hibernate or Hibernate), use this command to force the USB physical interface (PHY) to shut down. **Password required:** No **Notes:** • This command can be used only when the +KSLEEP <level> = 1 (Lite Hibernate) or 2 (Hibernate), otherwise the command returns an error. • To reconnect the USB PHY, trigger (toggle) a Low Power Mode wakeup event using a modem wakeup source (e.g., a wakeup pin or POWER_ON_N). For wakeup source details, refer to the RC71xx Product Technical Specification. **Usage:** • Execution: **AT!USBSHUTDOWN** Response: None Purpose: Force the USB PHY to disconnect and shutdown. No response is received because the USB connection is down. **Parameters:** None **Example(s):** • Module is currently not in Low Power Mode: AT!USBSHUTDOWN ← *Command will fail because the module is not in Low Power Mode (i.e., +KSLEEP level 1 or level 2)* +CME ERROR: 4 AT+KSLEEP=0,1,5000 OK AT!USBSHUTDOWN ← *USB PHY shuts down successfully*

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+WESHDOWN	**Emergency Shutdown** Use this command to manually trigger an emergency shutdown, or to enable (or disable) the emergency shutdown feature, which allows a GPIO to trigger the shutdown. **Password required:** No **Persistent across power cycles:** Yes **Notes:** • This command performs a fast power down without an IMSI detach request being sent to the network. • Only one GPIO at a time can be configured for emergency shutdown. **Usage:** • Execution: **AT+WESHDOWN=<mode>[,<gpio_index>]** Response: OK Purpose: Disable/enable emergency shutdown by GPIO, or trigger a shutdown directly. • Query: **AT+WESHDOWN?** Response: +WESHDOWN: <mode>,<gpio_index> OK Purpose: Report the current emergency shutdown GPIO setting. • Query List: **AT!WESHDOWN=?** Purpose: Display valid execution format and parameter values. **Parameters:** <mode> • 0—Disable emergency shutdown feature by GPIO • 1—Enable emergency shutdown feature by GPIO • 2—Trigger an emergency shutdown <gpio_index> (GPIO used as input to trigger emergency shutdown on the falling edge) • Valid values: 2, 4, 21, 42

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+WFWUPD	**Download and install firmware delta package locally over AT port** This command is used to download and install the firmware delta package locally over the AT port using hex-format data. **Password required:** Yes (see !ENTERCND for details) **Notes:** • After AT+WFWUPD=0 is sent to start the package download, no reset is made during the package download. **Usage:** • Execution: **AT+WFWUPD=<op>[,<pkgsn>,<data_size>,<data_in_hex-_string>,<crc8>]** Response: OK *or* ERROR Purpose: Start the requested operation (<op>). If <op>=0, OK indicates the device is ready to receive the firmware package, and ERROR indicates the initial download failed. If <op>=1, OK indicates the fragments were received correctly and written successfully, and ERROR indicates the fragments were not received correctly or the write failed. If <op>=2, OK indicates the package is available and device will reboot immediately to start the firmware update, and ERROR indicates no package is available to be installed. • Query: **AT+WFWUPD?** Response: +WFWUPD: <pkg> OK Purpose: See the package loading status. • Query List: **AT+WFWUPD=?** Response: +WFWUPD: (list of supported <op>s) OK Purpose: Display a list of supported operations. **Parameters:** <op> (Operation mode) • Valid values: • 0—Start the firmware package download. • 1—Send the fragments of firmware package. • 2—Install the firmware update from the downloaded package. <pkgsn> (Package fragment serial number) • Decimal • Valid range: 0 to (total fragment size - 1) <data_size> (Package fragment payload data size, in bytes) • Decimal • Maximum: 1024 <data_in_hex_string> (Package fragment payload data, in bytes) • Hexadecimal string (Continued on next page)

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+WFWUPD (continued)	**Download and install the firmware package locally over AT port (continued)** <crc8> (Package fragment 8-bit cyclic redundancy check value) • Hexadecimal <pkg> (Package loading status) • Hexadecimal ASCII value. • Valid values: • 0—No package is available to be installed • 1—Package is loaded and available to be installed **Examples:** • AT+WFWUPD=? +WFWUPD: (0-2) OK • AT+WFWUPD? +WFWUPD: 0 //No package is loaded OK • AT+WFWUPD=0 // Start the firmware package download OK // The device is ready to receive the package fragment AT+WFWUPD=1,0,564,c8000004... 447fb0ff,47 // Send package fragments to device • AT+WFWUPD? +WFWUPD: 1 //Package is loaded and ready to install OK • AT+WFWUPD=2 //Launch the firmware update from the package OK //device reset and then start the package update • +WFWUPD: 0 //After bootup, notification is shown that firmware update from the package is successful

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+WFWUPD (notification)	**Package install is launched—Unsolicited notification** Unsolicited notification received after package install is launched with AT+WFWUPD=1 The update status <stat> is retrieved from AT!BCFWUPDATESTATUS **Notification format:** **+WFWUPD: <stat>** **Parameters:** <stat> (Update status) • 0 - Bootloader reports successful installation • 1 - Package installation result fails or status is unknown *Note: For details, refer to !BCFWUPDATESTATUS for the firmware update status, ATI3 and ATI8 for the updated version information.*

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!WIFISCAN	**Scan for available Wi-Fi networks** Search for available Wi-Fi networks. **Password required:** No **Usage:** • Execution: **AT!WIFISCAN** ← *Search with all parameters set to default values.* *or* **AT!WIFISCAN=[<time>][,[<round>][,[<maxbssidnum>][,[<scantimeout>][,[<priority>][,[<channelRecLen>][,[<channelCount>][,[<channelId1>][,[<channelId2>][,[<channelId3>][,[<channelId4>][,[<channelId5>][,[<channelId6>][,[<channelId7>][,[<channelId8>][,[<channelId9>][,[<channelId10>][,[<channelId11>][,[<channelId12>][,[<channelId13>][,[<channelId14>]]]]]]]]]]]]]]]]]]]]]** ↑ *Note: Do* <u>*not*</u> *include more <channelId#> parameters than the <channelCount>, otherwise a CME ERROR will be returned.* Response: !WIFISCAN:(<ecn>,<ssid>,<rssi>,<mac>,<channel>) [...] OK Purpose: Search for available Wi-Fi networks that satisfy the specified search parameters. Note: Default values (see parameter descriptions below) will be used for any parameters that are not specified. • Query: **AT!WIFISCAN?** Response: !WIFISCAN:<time>,<round>,<maxbssidnum>,<scantimeout>,<priority>,<channelRecLen>,<channelCount>,<channelId1>[,<channelId2>[,<channelId3>[,<channelId4>[,<channelId5>[,<channelId6>[,<channelId7>[,<channelId8>[,<channelId9>[,<channelId10>[,<channelId11>[,<channelId12>[,<channelId13>[,<channelId14>]]]]]]]]]]]]] OK Purpose: Display the parameter values that were used for the most recent Wi-Fi scan. (Note: The number of <channelId#> fields in the response depends on the <channelCount> value.) • Query List: **AT!WIFISCAN=?** Purpose: Return the execution command format and the supported parameter values. **Parameters:** <time> (Maximum scan response time, in ms) • Valid values: 4000–255000 • Default: 12000 <round> (Maximum number of Wi-Fi scan rounds) • Valid values: 1–3 • Default: 1 <maxbssidnum> (Maximum number of Wi-Fi 2.4 GHz BSSIDs to scan for) • Valid values: 4–10 • Default: 5 (Continued on next page)

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!WIFISCAN (continued)	**Scan for available Wi-Fi networks (continued)** <scantimeout> (Maximum search time per Wi-Fi scan round, in seconds) • Valid values: 1–255 • Default: 5 <priority> (Wi-Fi scan priority) • Valid values: • 0—Data preferred. If the UE is in connection state (i.e., has a data connection), the Wi-Fi scan procedure will not start until the UE returns to the Idle state. • 1—Wi-Fi preferred. If the UE is in connection state (i.e., has a data connection), the UE releases the connection immediately and starts the Wi-Fi scan procedure. • Default: 0 <channelRecLen> (Maximum scan time for each specified 2.4 GHz <channelID_n>, in ms) • Valid values: 100–280 • Default: 280 <channelCount> (Number of 2.4 GHz frequency channels to scan) • Valid values: • To scan specific channels in a specific order: 1–14. Set <channelCount> to the number of channels to scan, and specify the channels to scan in the first <channelId*x*> parameters. (e.g., to scan five specific channels, set <channelCount>=5 and set the channel numbers in <channelId1>..<channelId5>) • To scan all channels, set <channelCount> to ’1’, and set <channelId1> to ’0’. • Default: 1 <channelId1> ... <channelId14> (Wi-Fi 2.4 GHz frequency channel to scan) • Valid values: • To scan specific channels in a specific order: 1–14. Set <channelCount> to the number of channels to scan, and specify the channels to scan in the <channelId*x*> parameters. • To scan all channels, set <channelCount> to ’1’, and set <channelId1> to ’0’. • Default: 0 <ecn> (Reserved parameter) • This parameter is unused. The value is always displayed as ’-’ (quotation marks are not displayed). <ssid> (Name of a discovered Wi-Fi 2.4 GHz network) • ASCII string, including quotation marks • If no name is broadcast, the value is displayed as ’-’ (quotation marks are not displayed) <rssi> (Discovered Wi-Fi network’s received signal strength level, in dBm) • Integer • e.g., -59 <mac> (Discovered Wi-Fi network’s MAC address) • ASCII string, including quotation marks • Format: "FF:FF:FF:FF:FF:FF" <channel> (Discovered Wi-Fi network’s available channel) • Valid values: 1–14 (Continued on next page)

Table 3-2: Modem Status Command Details (Continued)

Command	Description
!WIFISCAN (continued)	**Scan for available Wi-Fi networks (continued)** **Example(s):** • Scan using default values for all parameters: **at!wifscan** !WIFISCAN:(-,-,-57,"9C:8C:D8:A6:0E:26",1) !WIFISCAN:(-,"MyWiFi",-57,"9C:8C:D8:A6:0E:21",1) !WIFISCAN:(-,-,-71,"9C:8C:D8:A6:E1:A0",1) !WIFISCAN:(-,"Public hotspot"-,-82,"48:22:54:EB:FA:DE",2) !WIFISCAN:(-,-,-85,"D8:FE:E3:5F:5D:38",1) OK **at!wifiscan?** !WIFISCAN:12000,1,5,5,0,280,1,0 • Scan for available Wi-Fi networks on channels **9, 5, 6, 1, 3, 7** when the UE is in Idle mode (**<priority>=0**; i.e., if the UE has a data connection, do not start the scan until the UE returns to Idle mode): **at!wifiscan=12000,1,5,3,0,280,6,9,5,6,1,3,7** !WIFISCAN:(-,"Guest wifi",-46,"A0:36:BC:36:C8:28",8) !WIFISCAN:(-,-,-47,"F8:32:E4:79:E4:F8",8) !WIFISCAN:(-,-,-81,"08:BF:B8:4A:2F:C2",9) !WIFISCAN:(-,-,-86,"72:87:42:5C:85:40",6) !WIFISCAN:(-,-,-87,"68:28:CF:FD:94:D2",6) OK • Do the same scan as above, but use too many <channelId#> parameters. In this example, the <channelCount>=6, but there are eight <channelID#> parameters, so "+CME ERROR: 50" is returned: **at!wifiscan=12000,1,5,3,0,280,6,9,5,6,1,3,7,9,4** +CME ERROR: 50 • Scan for available Wi-Fi networks on any channels (**<channelCount>=1**; **<channelID1>=0**) immediately (**<priority>=1**; i.e., if the UE has a data connection, temporarily release the connection and enter Idle mode before starting the scan): **at!wifiscan=12000,1,5,5,1,280,1,0** !WIFISCAN:(-,-,-58,"C2:56:4A:14:41:1C",1) !WIFISCAN:(-,-,-61,"68:28:CF:FF:AD:42",1) !WIFISCAN:(-,-,-68,"E2:B0:FC:CA:33:67",1) !WIFISCAN:(-,"XYZ_access_point",-71,"B4:B0:24:F9:EF:23",3) !WIFISCAN:(-,-,-81,"9C:5C:F9:2B:39:04",1) OK

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+WUSLMSK	**Enable/disable unsolicited notifications** Enable or disable unsolicited notifications. When enabled, unsolicited notifications are output to the AT port when specific events occur. By default, unsolicited notifications are disabled. **Password required:** No **Reset required to apply changes:** No **Persistent across power cycles:** Yes **Usage:** • Execution: **AT+WUSLMSK=<bitmask>,<mask_position>** Response: OK Purpose: Enable or disable the selected notifications (in <bitmask>) defined in the specified 32-bit <mask_position>. • Query: **AT+WUSLMSK?** Response: +WUSLMSK: <bitmask><mask_position> OK Purpose: Report current state of system mode indications (enabled/disabled), showing the upper 32-bit mask followed by the lower 32-bit mask. Example: +WUSLMSK: 00002B0E710241D0 OK (The upper mask is 00002B0E, and lower mask is 710241D0.) • Query List: **AT+WUSLMSK=?** Purpose: Return the execution command format. See the parameter descriptions below for details. **Parameters:** <bitmask> (Unsolicited notifications bit mask, applied to the specified 32-bit <mask_position>) • Bit mask indicating which notifications to enable/disable. • Range: 00000000–FFFFFFFF. For example: • 00000000=All bits off (Default value) • FFFFFFFF=All bits on • Any other combination=Combination of bits off and on • See **LOWER unsolicited notifications mask** on page 75 and **UPPER unsolicited notifications mask** on page 76 for supported messages <mask_position> (The 32-bit mask of notifications that the <bitmask> is to be applied to.) • 0=Lower 32-bit mask • 1=Upper 32-bit mask (Continued on next page)

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+WUSLMSK (continued)	**Enable/disable unsolicited notifications (continued)** *Note: Notification support is firmware-dependent. Some of these notifications may not be supported or applicable.* **LOWER unsolicited notifications mask** Bit Mask value Unsolic. Notif. Responsible for: 0 0x00000001 --- Reserved 1 0x00000002 --- Reserved 2 0x00000004 --- Reserved 3 0x00000008 --- Reserved 4 0x00000010 --- Reserved 5 0x00000020 --- Reserved 6 0x00000040 --- Reserved 7 0x00000080 --- Reserved 8 0x00000100 --- Reserved 9 0x00000200 --- Reserved 10 0x00000400 --- Reserved 11 0x00000800 --- Reserved 12 0x00001000 --- Reserved 13 0x00002000 --- Reserved 14 0x00004000 --- Reserved 15 0x00008000 --- Reserved 16 0x00010000 --- Reserved 17 0x00020000 --- Reserved 18 0x00040000 --- Reserved 19 0x00080000 --- Reserved 20 0x00100000 --- Reserved 21 0x00200000 --- Reserved 22 0x00400000 --- Reserved 23 0x00800000 --- Reserved 24 0x01000000 --- Reserved 25 0x02000000 --- Reserved 26 0x04000000 --- Reserved 27 0x08000000 --- Reserved 28 0x10000000 --- Reserved 29 0x20000000 --- Reserved 30 0x40000000 --- Reserved 31 0x80000000 --- (Continued on next page)

Table 3-2: Modem Status Command Details (Continued)

Command	Description
+WUSLMSK (continued)	**Enable/disable unsolicited notifications (continued)** **UPPER unsolicited notifications mask** *Note: Notification support is firmware-dependent. Some of these notifications may not be supported or applicable.* Bit / Mask value / Unsolic. Notif. / Responsible for: 0 / 0x00000001 / --- / Reserved 1 / 0x00000002 / !PCVOLT / PMIC voltage state change 2 / 0x00000004 / !PCTEMP / PMIC temperature state change 3 / 0x00000008 / !PATEMP / PA Temperature state change 4 / 0x00000010 / --- / Reserved 5 / 0x00000020 / --- / Reserved 6 / 0x00000040 / --- / Reserved 7 / 0x00000080 / --- / Reserved 8 / 0x00000100 / --- / Reserved 9 / 0x00000200 / --- / Reserved 10 / 0x00000400 / --- / Reserved 11 / 0x00000800 / --- / Reserved 12 / 0x00001000 / --- / Reserved 13 / 0x00002000 / --- / Reserved 14 / 0x00004000 / --- / Reserved 15 / 0x00008000 / --- / Reserved 16 / 0x00010000 / --- / Reserved 17 / 0x00020000 / --- / Reserved 18 / 0x00040000 / --- / Reserved 19 / 0x00080000 / --- / Reserved 20 / 0x00100000 / --- / Reserved 21 / 0x00200000 / --- / Reserved 22 / 0x00400000 / --- / Reserved 23 / 0x00800000 / --- / Reserved 24 / 0x01000000 / --- / Reserved 25 / 0x02000000 / --- / Reserved 26 / 0x04000000 / --- / Reserved 27 / 0x08000000 / --- / Reserved 28 / 0x10000000 / --- / Reserved 29 / 0x20000000 / --- / Reserved 30 / 0x40000000 / --- / Reserved 31 / 0x80000000 / --- / Reserved

4: Diagnostic Commands

4.1 Introduction

This chapter describes commands used to diagnose modem problems.

4.2 Command summary

The table below lists the commands described in this chapter.

Table 4-1: Diagnostic Commands

Command	Description	Page
!BCFWUPDATESTATUS	Report status of most recent firmware update attempt	78

4.3 Command reference

Table 4-2: Diagnostic Command Details

Command	Description
!BCFWUPDATESTATUS	**Report status of most recent firmware update attempt** Return the status of the most recent firmware update attempt made since the last cold restart. **Password required:** No **Usage:** • Execution: **AT!BCFWUPDATESTATUS** Response: !BCFWUPDATESTATUS: <result> *or* !BCFWUPDATESTATUS: <result> Failed IMG TYPE <type>, DATA <data>, PART <part> OK Purpose: Return the status of the most recent firmware update attempt. The second response format appears only if <result> = “FAILED”. **Parameters:** <result> (Status of last firmware update attempt) • ASCII string: • “UNKNOWN”—Status of last attempt is unknown. • “SUCCESS” —Last update was successful. • “FAILED”—Last update failed. <type> (Firmware image type that failed to update) • ASCII string <data> (Reference data for failed image) • Location of the reference data as an offset in the CWE image • Valid range: 0–(2^{32}-1) <part> (Partition associated with the failed image) • ASCII string

5: Test Commands

5.1 Introduction

To obtain regulatory approval and carrier approvals for your product, you may be required to perform tests on the radio component of the embedded modem. This chapter describes AT commands used to perform those tests.

Warning: *These commands are intended for use by developers, not end-users. The commands should be used only in a controlled network environment.*

In most cases the modem must be in a particular mode before you can issue the AT commands to perform particular tests. Therefore, the order in which you issue certain commands is important. Three AT commands are important in setting the mode:

- !DAFTMACT—puts the modem in factory test mode (a non-signaling mode). You must issue **AT!DAFTMACT** before issuing +WMRXPOWER or +WMTXPOWER.

5.2 Command summary

The table below lists the commands described in this chapter.

Table 5-1: Test Commands

Command	Description	Page
!DAFTMACT	Put modem into Factory Test Mode	81
!DAFTMDEACT	Put modem into Online Mode from Factory Test Mode	82
!LEDPATTERN	Test LED	83
+WMRXPOWER	Run LTE Rx Power Test	84
+WMTXPOWER	Run LTE Tx Power Test	86

5.3 Command reference

Table 5-2: Test Command Details

Command	Description
!DAFTMACT	**Put modem into Factory Test Mode** Place the modem in FTM (Factory Test Mode). FTM is a non-signaling mode that allows the radio component to be manually configured to conduct certain types of tests. The modem must be in FTM mode to use the test commands described in this chapter (except for commands that start with “!DACGPS” **Notes:** • When this command executes successfully, the modem responds with the value 290300. Any other response indicates an error. **Password required:** Yes (see !ENTERCND for details) **Usage:** • Query: **AT!DAFTMACT** Response: 290300 *(Success. Any other response indicates an error.)* OK Purpose: Place modem in FTM mode (from online mode)

Table 5-2: Test Command Details (Continued)

Command	Description
!DAFTMDEACT	**Put modem into Online Mode from Factory Test Mode** Take the modem out of FTM and put it back into online mode. (!DAFTMACT puts the modem into FTM.) **Notes:** • When this command executes successfully, the modem responds with the value 290400. Any other response indicates an error. **Password required:** Yes (see !ENTERCND for details) **Usage:** • Query: **AT!DAFTMDEACT** Response: 290400 *(Success. Any other response indicates an error.)* OK Purpose: Place modem in online mode (from FTM mode).

Table 5-2: Test Command Details (Continued)

Command	Description
!LEDPATTERN	**Test LED** Test an LED by turning it on (light), off (dark), or set an on/off pattern (pulse width modulation (pwm). When finished testing the LED, reboot the device to return to normal LED operation. **Password required:** Yes (see !ENTERCND for details) **Usage:** • Execution: **AT!LEDPATTERN=<mode>[,<pwm_duration>,<pwm_cycle>]** Response: OK Purpose: Turn the LED on (light) or off (dark), or set a pwm on/off pattern. • Query List: **AT!LEDPATTERN=?** Purpose: Display the assignment command format and valid parameter options. **Parameters:** <mode> (LED mode • 0—Off • 1—On • 2—PWM (Pulse Width Modulation) <pwm_duration> (Percentage of <pwm_cycle> that the LED is on) • This parameter is used only when <mode>=2. • Decimal <pwm_cycle> (Duration of a single on/off sequence, in ms) • This parameter is used only when <mode>=2. • Decimal **Example(s):** AT!LEDPATTERN=0 ← Turn LED off OK AT!LEDPATTERN=1 ← Turn LED on OK AT!LEDPATTERN=2,40,200 ← Set PWM cycle to 200ms—On (80 ms)/ Off (120 ms) OK AT!LEDPATTERN=2,40,2000 ← Set PWM cycle to 2 seconds—On (800 ms)/ Off (1200 ms) OK

Table 5-2: Test Command Details (Continued)

Command	Description
+WMRXPOWER	**Run LTE Rx Power Test** This command starts/stops the Rx power test for a specified band/channel. **Password required:** Yes **Reset required to apply changes:** No **Usage Requirements:** Before this command can be used: • Use !DAFTMACT to enter FTM mode. **Notes:** • If the module is registered to a network when this command (or +WMTXPOWER) is used with <enable>=1, the module will disconnect from the network and enter RF test mode. • This command is not available when +WMTXPOWER is enabled. **Usage:** • Execution: **AT+WMRXPOWER=<enable>[,<band>,<channel>]** Response: +WMRXPOWER: <power1> OK Purpose: Start/stop the Rx power test. • Query: **AT+WMRXPOWER?** Response: +WMRXPOWER: <enable>[,<band>,<channel>] OK Purpose: Display the current Rx power test state (stopped/started). • Query List: **AT+WMRXPOWER=?** Purpose: Return the supported parameter values. **Parameters:** <enable> (Start/stop Rx measurement) • Valid values: • 0—Stop Rx measurement • 1—Start Rx measurement • Note—When <enable>=0, no other parameters are used in the execution command format. When <enable>=1, all parameters are required. <band> (Rx band to read) • In the execution command format, this parameter is only used when <enable>=1. • Valid values: • 1—Band 1 (EU SKU only) • 2—Band 2 (NA SKU only) • 3—Band 3 (EU SKU only) • 4—Band 4 (NA SKU only) • 7—Band 7 (EU SKU only) • 8—Band 8 (EU & NA SKUs) • 12—Band 12 (NA SKU only) • 13—Band 13 (NA SKU only) • 20—Band 20 (EU SKU only) • 28—Band 28 (EU SKU only) • 66—Band 66 (NA SKU only) • 71—Band 71 (NA SKU only) (Continued on next page)

Table 5-2: Test Command Details (Continued)

Command	Description
+WMRXPOWER (continued)	**Run LTE Rx Power Test (continued)** <channel> (Rx channel to read) • In the execution command format, this parameter is only used when <enable>=1. • Valid values are based on the <band> value: • If <band>=1: 0–599 (EU SKU only) • If <band>=2: 600–1199 (NA SKU only) • If <band>=3: 1200–1949 (EU SKU only) • If <band>=4: 1950–2399 (NA SKU only) • If <band>=7: 2750–3449 (EU SKU only) • If <band>=8: 3450–3799 (EU & NA SKUs) • If <band>=12: 5010–5179 (NA SKU only) • If <band>=13: 5180–5279 (NA SKU only) • If <band>=20: 6150–6449 (EU SKU only) • If <band>=28: 9210–9659 (EU SKU only) • If <band>=66: 66436–67335 (NA SKU only) • If <band>=71: 68586–68935 (NA SKU only) <power1> (Received power at primary antenna, in dBm) • In the execution command format, this parameter is only used when <enable>=1. • e.g., -95.0 **Example(s):** `AT+WMRXPOWER=1,4,1950 // Read B4 Earfcn 1950` `+WMRXPOWER: -95.0 // Rx power -95.0 dBm` `OK` `AT+WMRXPOWER=0 //RX disable` `+WMRXPOWER: 0` `OK`

Table 5-2: Test Command Details (Continued)

Command	Description
+WMTXPOWER	**Run LTE Tx Power Test** This command starts/stops the Tx power test for a specified band/channel. **Password required:** Yes **Reset required to apply changes:** Yes **Usage Requirements:** Before this command can be used: • Use !DAFTMACT to enter FTM mode. **Notes:** • If the module is registered to a network when this command (or +WMRXPOWER) is used with <enable>=1, the module will disconnect from the network and enter RF test mode. • This command is not available when +WMRXPOWER is enabled. **Usage:** • Execution: **AT+WMTXPOWER=<enable>[,<band>,<channel>,<power_level>,<tx_type>]** Response: OK Purpose: Start/stop the Tx power test. • Query: **AT+WMTXPOWER?** Response: +WMTXPOWER: <enable>[,<band>,<channel>,<power_level>,<tx_type>] OK Purpose: Display the current Tx power test state (stopped/started). • Query List: **AT+WMTXPOWER=?** Purpose: Return the execution command format and the supported parameter values. **Parameters:** <enable> (Start/stop burst emission) • Valid values: • 0—Stop burst emission • 1—Start burst emission • Note—When <enable>=0, no other parameters are used in the execution command format. When <enable>=1, all parameters are required. (Continued on next page)

Table 5-2: Test Command Details (Continued)

Command	Description
+WMTXPOWER (continued)	**Set LTE Tx Power (continued)** <band> (Tx band for burst emission) • In the execution command format, this parameter is only used when <enable>=1. • Valid values: • 1—Band 1 (EU SKU only) • 2—Band 2 (NA SKU only) • 3—Band 3 (EU SKU only) • 4—Band 4 (NA SKU only) • 7—Band 7 (EU SKU only) • 8—Band 8 (EU & NA SKUs) • 12—Band 12 (NA SKU only) • 13—Band 13 (NA SKU only) • 20—Band 20 (EU SKU only) • 28—Band 28 (EU SKU only) • 66—Band 66 (NA SKU only) • 71—Band 71 (NA SKU only) <channel> (Tx channel for burst emission) • In the execution command format, this parameter is only used when <enable>=1. • Valid values are based on the <band> value: • If <band>=1: 18000–18599 (EU SKU only) • If <band>=2: 18600–19199 (NA SKU only) • If <band>=3: 19200–19949 (EU SKU only) • If <band>=4: 18000–18599 (NA SKU only) • If <band>=7: 18000–18599 (EU SKU only) • If <band>=8: 18000–18599 (EU & NA SKUs) • If <band>=12: 18000–18599 (NA SKU only) • If <band>=13: 18000–18599 (NA SKU only) • If <band>=20: 18000–18599 (EU SKU only) • If <band>=28: 18000–18599 (EU SKU only) • If <band>=66: 18000–18599 (NA SKU only) • If <band>=71: 18000–18599 (NA SKU only) <power_level> (Absolute output power, in dBm ×100) • In the execution command format, this parameter is only used when <enable>=1. • Valid range: 0–2600, for all bands (e.g., 0 dBm – 2600 dBm) <tx_type> (Transmitted signal type) • In the execution command format, this parameter is only used when <enable>=1. • Valid values: • 1—CW (continuous waveform). For customers who use a spectrum analyzer (i.e., a CMW tester is not available). **Example(s):** `AT+WMTXPOWER=1,2,18600,2300,1` // B2 Earfcn 18600 with a power // level of 23dbm Tx type(CW) `OK` `AT+WMTXPOWER=0` //RX disable `+WMTXPOWER: 0` `OK`

6: Memory Management Commands

6.1 Introduction

The modem uses non-volatile memory to store:

- Factory calibration data
- Settings made in a host application such as Skylight.

The commands in this chapter allow you to back up and restore the data in non-volatile memory.

6.2 Command summary

The table below lists the commands described in this chapter:

Table 6-1: Memory Management Commands

Command	Description	Page
!RMARESET	Restore device	89

6.3 Command reference

Table 6-2: Memory Management Command Details

Command	Description
!RMARESET	**Restore device** Restore the device to its original provisioned (OEM default) state, or to the latest backed-up state. **Notes:** • The module does not reboot automatically. It must be manually rebooted to use the restored settings. **Password required:** Yes (see !ENTERCND for details) **Usage:** • Execution: **AT!RMARESET=<level>** Response: !RMARESET: DEVICE REBOOT REQUIRED Items Restored: ##### Items Deleted: ##### Items Defaulted: ##### Items Skipped: ##### OK Purpose: Restore device to the requested state. **Parameters:** <level> (Restoration type) • 1=Default OEM provisioned state • 3=Latest backed-up state

7: SIM Commands

7.1 Introduction

This chapter describes commands used to communicate with an installed SIM.

7.2 Command summary

Table 7-1 lists the commands described in this chapter:

Table 7-1: SIM Commands

Command	Description	Page
+CCHC	Close a Local Channel by Session ID	91
+CCHO	Open a Local Channel and Return the Session ID	92
+CCID	Return SIM ICCID	93
+CGLA	Send APDU Command to SIM Card	94
+CPINR	Display remaining number of SIM unlock retries	95
+CSPN	Display SIM card service provider's name (SPN)	96
!UIMS	Select active UIM interface	97

7.3 Command reference

Table 7-2: SIM Command Details

Command	Description
+CCHC	**Close a Local Channel by Session ID** **Password required:** **Usage:** • Execution: **AT+CCHC= <session_id>** Response: +CCHC OK *or* +CME ERROR: <err> Purpose: Close a local channel. • Query: **AT+CCHC=?** Response: OK Purpose: Query list supported. **Parameters:** <session_id> (Integer) • Session ID to target a specific application on the USIM using logical channels mechanisms.

Table 7-2: SIM Command Details (Continued)

Command	Description
+CCHO	**Open a Local Channel and Return the Session ID** **Notes:** The +CCHO execute command gives the <session_id> when it receives SIM application response status words as shown below: • '90' '00'—normal ending of the command • '91' 'XX'—normal ending of the command with extra information from the proactive UICC containing a command for the terminal length 'XX' of the response data • '92' 'XX'—normal ending of the command with extra information concerning an ongoing data transfer session **Password required:** No **Usage:** • Execution: **AT+CCHO=<dfname>** Response: <session_id> OK *or* +CME ERROR: <err> Purpose: Open a local channel • Query: **AT+CCHO=?** Response: OK Purpose: Query list supported **Parameters:** <dfname> (Integer) • All selectable applications in the UICC are referenced by a DF name coded on 1 – 16 bytes <session_id> (Integer) • Session ID to target a specific application on the USIM using logical channels mechanisms.

Table 7-2: SIM Command Details (Continued)

Command	Description
+CCID	**Return SIM ICCID** Return the active SIM's ICCID. **Password required:** No **Usage:** • Query: **AT+CCID?** *or* **AT+CCID** Response: +CCID: <iccid> OK *or* +CME ERROR: <error> Purpose: Display the ICCID of the active SIM. **Parameters:** <iccid> (ICCID of the SIM/eUICC currently being tested): • 20 digit decimal number—This number is often printed on the SIM card.

Table 7-2: SIM Command Details (Continued)

Command	Description
+CGLA	**Send APDU Command to SIM Card** **Notes:** • When invalid parameter is given, an empty response is returned: AT+CGLA=257,14,"TW010100002100" //invalid parameter +CGLA: 0,"" OK **Password required:** **Usage:** • Execution: **AT+CGLA= <sessionid>, <length>, <command>** Response: +CGLA: <length>,<response> OK *or* +CME ERROR: <err> Purpose: Sends APDU command to SIM card. **Parameters:** <session_id> (Integer) • Used as the identifier of the session to be used in order to send the APDU commands to the UICC. It is mandatory in order to send commands to the UICC when targeting applications on the smart card using a logical channel other than the default channel (channel "0"). <length (Integer) • Length of the characters that are sent to TE in <command> or <response> (two times the actual length of the command or response). <command> (String) • Command passed on by the MT to the UICC in the format as described in 3GPP TS 31.101 in hexadecimal format (refer to +CSCS). <response> (String) • Response to the command passed on by the UICC to the MT in the format as described in 3GPP TS 31.101 in hexadecimal format (refer to +CSCS).

Table 7-2: SIM Command Details (Continued)

Command	Description
+CPINR	**Display remaining number of SIM unlock retries** Display the number of remaining SIM unlock retries. **Password required:** No **Usage:** • Execution: **AT+CPINR=<CPIN TYPE>** Response: +CPINR: <CPIN TYPE>,<remaining> OK Purpose: Display the number of remaining retries for the specified PIN/PUK type. • Query: **AT+CPINR** Response: +CPINR: SIM PIN,<remaining> +CPINR: SIM PUK,<remaining> +CPINR: SIM PIN2,<remaining> +CPINR: SIM PUK2,<remaining> +CPINR: PH-FSIM PIN,<remaining> +CPINR: PH-NET PIN,<remaining> +CPINR: PH-NETSUB PIN,<remaining> +CPINR: PH-SP PIN,<remaining> +CPINR: PH-CORP PIN,<remaining> +CPINR: PH-FSIM PUK,<remaining> +CPINR: PH-NET PUK,<remaining> +CPINR: PH-NETSUB PUK,<remaining> +CPINR: PH-SP PUK,<remaining> +CPINR: PH-CORP PUK,<remaining> OK Purpose: Display the number of remaining retries for all PIN/PUK types. **Parameters:** <CPIN TYPE> (PIN/PUK type) ASCII string enclosed within quotes. Valid values: (Note: If there are any errors in this list, use AT+CPINR to display the full listof available types.) • “SIM PIN” • “SIM PUK” • “SIM PIN2” • “SIM PUK2” • “PH-FSIM PIN” • “PH-NET PIN” • “PH-NETSUB PIN” • “PH-SP PIN” • “PH-CORP PIN” • “PH-FSIM PUK” • “PH-NET PUK” • “PH-NETSUB PUK” • “PH-SP PUK” • “PH-CORP PUK” <remaining> (Number of retries remaining for specified PIN/PUK type) • 0–255 (maximum value is type-dependent)

Table 7-2: SIM Command Details (Continued)

Command	Description
+CSPN	**Display SIM card service provider's name (SPN)** Display the service provider name for the SIM card. **Password required:** No **Usage:** (Note: Execution and Query formats return the same response.) • Execution: **AT+CSPN** Response: +CSPN: <spn> OK *or* +ERROR Purpose: Display the SIM card's service provider name. • Query: **AT+CSPN?** Response: +CSPN: <spn> OK *or* +ERROR Purpose: Display the SIM card's service provider name. • Query List: **AT+CSPN=?** Response: OK Purpose: None. **Parameters:** <spn> (Service provider name): • ASCII string enclosed within quotes.

Table 7-2: SIM Command Details (Continued)

Command	Description
!UIMS	**Select active UIM interface** On a module that supports multiple UIM interfaces, select the active UIM interface. **Password required:** No **Persistent across power cycles:** Yes **Usage Requirements:** • The module must be in the modem power off state (AT+CFUN=0) before using this command to select an interface. • To enable or disable the UIM2 interface, use the !CUSTOM UIM2ENABLE customization. See !CUSTOM on page 32. **Notes:** • Recommendation: Because the RC71XX does not support SIM hotplug, good practice is to put the module in the modem power off state (AT+CFUN=0) before removing a SIM. **Usage:** • Execution: **AT!UIMS=<uim>** Response: OK *or* ERROR Purpose: Configure the module to use the selected UIM interface. • Query: **AT!UIMS?** Response: !UIMS: <uim> OK Purpose: Display the currently selected interface. • Query List: **AT!UIMS=?** Response: !UIMS: (List of supported <uim>s) OK Purpose: Return the command format and the supported parameter values. **Parameters:** <uim> (SIM interface): • Valid values: ◦ 0—UIM1. External UIM interface #1 ◦ 1—UIM2. External UIM interface #2 • Note—Depending on the module, the interface may be exposed to an external SIM connector. **Example(s):** `AT+CFUN=0` `OK` `AT!CUSTOM="UIM2ENABLE",1` `OK` `AT!UIMS=1` `OK` `AT!RESET` *or* `AT+CFUN=1` `OK`

8: I/O Commands

8.1 Introduction

This chapter describes commands used to configure and manage GPIOs, ADCs and other IOs.

8.2 Command summary

Table 8-1 lists the commands described in this chapter.

Table 8-1: I/O Commands

Command	Description	Page
!MADC	Displays ADC values	99
+WIOCFG	GPIO Configuration	100
+WIOR	Read GPIO Value	102
+WIOW	Write GPIO Value	103
+WRID	Set/query Ring Indicator Duration	104
+WWAKESET	Set/query Wake Up Event Mask	105

8.3 Command reference

Table 8-2: I/O Command Details

Command	Description
!MADC	**Displays ADC values** Reads one of the available ADCs (Analog to Digital Converters) **Notes:** • MADC does the unit conversion and displays the ADC value with units as mV, V, Celcius etc, whereas AT!ADC only displays the raw value of ADC sensor. **Password required:** No **Usage:** • Execution: **AT!MADC=<input_adc>** Response: !MADC: <value> OK Purpose: Show the value being reported by the specified ADC. • Query List: **AT!MADC=?** Response: AT!MADC=<input_adc> input adc: 0: VBATT 1: PMIC_THERM 2: PA_THERM 3: ADC1 4: ADC0 OK Purpose: Displays the valid execution format and parameter values. **Parameters:** <input_adc> (Analog to Digital Converters) • Integer • 0—VBATT (Battery voltage) • 1—PMIC_THERM (Power Management Integrated Circuit Thermistor) • 2—PA_THERM (Power Amplifier Thermistor) • 3—ADC1 • 4—ADC0 <value> (Value returned from ADC) • ASCII string • Note: Contents depend on the ADC being polled.

Table 8-2: I/O Command Details (Continued)

Command	Description
+WIOCFG	**GPIO Configuration** Configure a specific GPIO (I/O port) for one of the following uses (indicated by the <func> parameter): • GPIO, accessible via AT commands (<func> = 4) • Deallocate port (<func> = 0) **Notes:** • To enable 'Reset Out', set <gpio> = 6 and <func> = 0. Refer to the RC71xx Product Technical Specification for details. **Password required:** No **Reset required to apply changes:** No **Persistent across power cycles:** Yes **Usage:** • Execution (for <func>=0; Mark GPIO as unallocated): **AT+WIOCFG=<gpio>,<func>** Response: OK *(If the port configuration works as requested)* Purpose: Deallocate a GPIO. • Execution (for <func>=4; Allocate GPIO for General use or for Embedded Host use: **AT+WIOCFG=<gpio>,<func>[,<dir>,<state>,<pull>,<trigger>]** Response: OK *(If the port configuration works as requested)* *or* ERROR *(If the port is already allocated—the current <func> value is not 0)* Purpose: Allocate the requested port (<idx>) for use as a GPIO or for control by the embedded host. • Query: **AT+WIOCFG?** Response: +WIOCFG: <gpio>,<func>,<dir>,<state>,<pull>,<trigger> ... +WIOCFG: <gpio>,<func>,<dir>,<state>,<pull>,<trigger> OK Purpose: Report the configuration for all ports. • Query List: **AT+WIOCFG=?** Purpose: Display valid execution format and parameter values. **Parameters:** <gpio> (Index of I/O port to be configured) • Valid range: 1–46. Use AT+WIOCFG? to view supported <gpio> values. • Example: AT+WIOCFG? +WIOCFG: 2,4,0,0,2,4 +WIOCFG: 4,4,0,0,2,5 +WIOCFG: 6,4,1,0,0,0 +WIOCFG: 7,4,1,0,0,0 ... The first parameters of each line of output are the valid <gpio> values (e.g. 2, 4, 6, 7, ...). • Note: To enable 'Reset Out', set <gpio> = 6 and <func> = 0. Refer to the RC71xx Product Technical Specification document for details. (Continued on next page)

Table 8-2: I/O Command Details (Continued)

Command	Description
+WIOCFG (continued)	**GPIO Configuration (continued)** <func> (I/O port usage) • 0—Unallocated • 4—General GPIO • Note: To enable 'Reset Out', set <gpio> = 6 and <func> = 0. <dir> (GPIO direction) • 0—Input • 1—Output <state> (Power-up state for external GPIO configured as an output) • 0—Output low level at power-up • 1—Output high level at power-up <pull> (Internal pull type for the I/O port) • 0—No pull • 1—Pull down • 2—Keeper • 3—Pull up <trigger> (Trigger type for I/O port configured as an input) • Note: <trigger> is not supported if <gpio>=6 (GPIO6) • 0—No trigger • 1—Trigger high • 2—Trigger low • 3—Trigger rising • 4—Trigger falling

Table 8-2: I/O Command Details (Continued)

Command	Description
+WIOR	**Read GPIO Value** Reads the pin value of a GPIO (General Purpose I/O port) that has been configured as an input. **Requirement(s):** Before using this command, AT+WIOCFG must be used to configure <gpio> as an output GPIO (i.e., <func>=4; <dir>=0). **Notes:** • This command returns an ERROR if the GPIO has been configured as an output. **Password required:** No **Usage:** • Execution: **AT+WIOR=<gpio>** Response: +WIOR: <value> OK *or (if <gpio> is configured as an output)* ERROR Purpose: Reads the specified GPIO's pin value. • Query List: **AT+WIOR=?** Response: +WIOR: (list of supported <gpio>s) OK Purpose: Display valid execution format and parameter values. **Parameters:** <gpio> (External GPIO number) • Valid range: 1—46 • Use +WIOCFG to view supported values. • Example: AT+WIOCFG? +WIOCFG: GPIO2 = 2,4,0,0,2,4 +WIOCFG: GPIO4 = 4,4,0,0,2,5 +WIOCFG: GPIO6 = 6,4,1,0,0,0 ... The first parameters of each line of output are the valid <gpio> values (e.g. 2, 4, 6, ...). <value> (GPIO pin value) • Integer • 0—Low level

Table 8-2: I/O Command Details (Continued)

Command	Description
+WIOW	**Write GPIO Value** Writes a GPIO (General Purpose I/O port) pin value. When using this command, check the AT+WIOCFG setting as below: • <func> = 4 (General GPIO) • <dir> = 1 (Output) **Password required:** No **Usage:** • Execution: **AT+WIOW=<gpio>,<value>** Response: OK Purpose: Write the specified GPIO's pin value. • Query List: **AT+WIOW=?** Response: +WIOW: (1-46),(0-1) OK Purpose: Display valid execution format and parameter values. **Parameters:** <gpio> (External GPIO number) • Note: Not all values are valid. Use +WIOCFG to view supported values. • Valid range: 1—42 <value> (GPIO pin value) • Valid range: 0—1 • 0—Low level • 1—High level

Table 8-2: I/O Command Details (Continued)

Command	Description
+WRID	**Set/query Ring Indicator Duration** Set or return the duration of the pulse that is asserted on the Ring Indicator line (pin RI1). (The pulse may be asserted under several different event conditions, but the pulse duration is the same.) Make sure to set the duration appropriately. While long durations may make sense for some events, it is possible that shorter events may expire before the pulse finishes (for example, an incoming call could expire or be re-routed to voicemail). The design is such that if an event expires before the pulse finishes, the wakeup reason and ring indicator will not be reset. **Password required:** No **Reset required to apply changes:** No **Persistent across power cycles:** Yes **Usage:** • Execution: **AT+WRID[=<n>]** Response: OK *or* ERROR *(If invalid assignment)* Purpose: Set the ring indicator pulse duration. If "=<n>" is not entered, the default pulse duration value (50 ms) is used. • Query: **AT+WRID?** Response: +WRID: <n> Purpose: Display the ring indicator pulse duration. • Query List: **AT+WRID=?** Purpose: Display valid execution format and parameter values. **Parameters:** <n> (Ring indicator pulse duration, in ms units) • 50–10000 (Default=50 ms). Range equates to 0.05–10.0 seconds. • Integer values only (pulse is set in 1 ms steps)

Table 8-2: I/O Command Details (Continued)

Command	Description
+WWAKESET	**Set/query Wake Up Event Mask** Set or query the WAKE mask setting, which indicates the actions that will generate a pulse on the Ring Indicator (RI1) output signal to "wake up" an application. The WAKE mask indicates all events that can generate the wake pulse. When an event occurs, the RI is asserted for the duration defined via AT+WRID and then de-asserts. If additional events occur while the RI is asserted, the RI is not re-asserted and the duration is not extended; it is assumed that the external processor is awakened by the first assertion. **Notes:** • Each time this command is used to set the mask, the previous setting is replaced. That is, the mask value must indicate all the events that will generate a pulse. **Password required:** No **Usage:** • Execution: **AT+WWAKESET[=<bitmask>]** Response: OK Purpose: Indicate which events pulse the RI pin. If "=<bitmask>" is not entered, the default mask value (16—Incoming SMS message) is used. • Query: **AT+WWAKESET?** Response: +WWAKESET: <bitmask> Purpose: Display the current mask value. • Query List: **AT+WWAKESET=?** Response: +WRID: (list of supported <bitmask>s) OK Purpose: Display valid execution format and parameter values. **Parameters:** <bitmask> (Events that will assert (pulse) the RI signal) • If more than one event will assert the signal, add the values. For example, to get notifications for both lost service and incoming voice calls, the <bitmask> value is 5. • 0—No notifications • 1—Lost service (for example, going from digital service to no service)—If the module is in deep sleep (32 kHz), the RI will assert and the module will remain asleep • 2—Service regained (going from no service to service)—If the module is in deep sleep (32 kHz), the RI will assert and the module will remain asleep. NOTE: Changing the SID and remaining on the same service type will NOT trigger the RI signal. • 16—Incoming SMS message (Default) • 32—Reserved • 64—Module restart (includes the first power up) • 128—Module has undergone Sudden Momentary Power Loss • 256—Reserved • 1024—Reserved • 4095—All events as listed above

9: Delta OTA Firmware Update Commands

9.1 Introduction

This chapter describes delta OTA frmware update related commands.

9.2 Command summary

Table 9-1 lists the commands described in this chapter.

Table 9-1: AirVantage Device Services Commands

Command	Description	Page
+WDSD	Device Services UART Local Download	107
+WDSI (notifications)	FOTA Service—Unsolicited notifications	109
+WDSR	Reply to firmware update request	110
+WDSTPF	Device Services Third-Party FOTA	111

9.3 Command reference

Table 9-2: Device Services Command Details

Command	Description
+WDSD	**Device Services UART Local Download** Download a firmware package locally by using the XMODEM protocol over the UART AT port. **Password required:** No **Notes:** • The response to the execution command is the letter "C" (repeating), which indicates the device is ready to receive data that is sent using the XMODEM protocol. • No reset occurs during the package download. • The download process times out if no data is sent by the host to the module within 5 minutes, and a +CME ERROR: 3 is returned. • When +WDSD completes (all data is received by the module), a +WDSE:3 notification will be received, requesting a user agreement to install the package. The only supported +WDSR reply is AT+WDSR=4 (accept the install)—installs cannot be delayed. **Usage:** • Execution: **AT+WDSD=<size>** Response: CCC...CCC ← *Indicates XMODEM is ready to send data* *[Host uses XMODEM to send data (<size> bytes)]* OK +WDSI: 3 ← *Notification indicates all data was received by the module and a user agreement is requested to install the package.* *or* +CME ERROR: <err> Purpose: Start a firmware download using XMODEM. • Query List: **AT+WDSD=?** Purpose: Return the execution command format and the supported parameter values. **Parameters:** <size> (Package size, in bytes) • Valid range: 1–491520 • File size limit is 480 KB **Example(s):** **AT+WDSD=256** CCCCCCCCCCC *[Host uses XMODEM to send data (1 Kb or 128 byte packets, depending on type)]* OK +WDSI: 3 **AT+WDSR=4** OK (Continued on next page)

Table 9-2: Device Services Command Details (Continued)

Command	Description
+WDSD (continued)	**Device Services UART Local Download (continued)** `+WDSI: 14` *[Wait for device to upgrade and reboot]* `+WDSI: 16`

Table 9-2: Device Services Command Details (Continued)

Command	Description
+WDSI (notification)	**FOTA Service—Unsolicited notifications** Unsolicited notifications received for various FOTA service events. **Notification format:** **+WDSI: <Event>[,<Data>]** **Parameters:** <Event> (FOTA Service event) • 0–2—Reserved • 3—Device has downloaded a package. The device requests a user agreement to install the downloaded package. The response can be sent using +WDSR (see +WDSR on page 110). • 4–9—Reserved • 10—Package was successfully downloaded and stored in flash. • 11–13—Reserved • 14—Update will be launched. • 15—OTA update client has finished unsuccessfully. • 16—OTA update client has finished successfully. • 17—Reserved • 18—Download progress: • No <Data> parameter—Download start • <Data> parameter—Percentage progress <Data> (Additional data for specific <Event>s) • (<Event>=18) Download progress: • Integer value (% complete) **Examples:** • +WDSI: 18,1 *1% of package has been downloaded* • +WDSI: 18, 100 *Entire package (100%) has been downloaded*

Table 9-2: Device Services Command Details (Continued)

Command	Description
+WDSR	**Reply to firmware update request** Reply to a user agreement request (see +WDSI (notification) on page 109 for details) from the module. **SIM card requirement:** No **Password required:** No **Usage:** • Execution: **AT+WDSR=<Reply>** Response: OK Purpose: Send <Reply> to a user agreement request from the module. • Query List: **AT+WDSR=?** Purpose: Display valid execution format and parameter values. **Parameters:** <Reply> (Reply type) • 4—Accept the install (install it now)

Table 9-2: Device Services Command Details (Continued)

Command	Description
+WDSTPF	**Device Services Third-Party FOTA** Configure an HTTP third-party FOTA package address and start the download. **Password required:** No **Persistent across power cycles:** Partial (<addr> configuration is persistent) **Notes:** • User agreements for firmware download and installation apply to the third-party FOTA service. +WDSR uses these user agreements to install packages. • +WDSI notifications will be received, indicating the different states of FOTA. **Usage:** • Execution (Set package URL): **AT+WDSTPF=<mode=0>,<addr>** Response: OK Purpose: Set the FOTA package's URL. This does not start the download. • Execution (Start FOTA operation): **AT+WDSTPF=<mode=1>** Response: OK Purpose: Start the FOTA operation using the currently-configured FOTA package URL. • Query: **AT+WDSTPF?** Response: +WDSTPF: 0,<addr> +WDSTPF: 1,<state> OK Purpose: Display the current configuration (address and state). • Query List: **AT+WDSTPF=?** Purpose: Return the execution command formats and the supported parameter values. **Parameters:** <mode> (Configuration option) • 0—Set the package URL using the <addr> parameter. • 1—Start the FOTA operation. When this mode is selected, download starts based on the user agreement configuration. <addr> (FOTA package address) • ASCII string; maximum length 255 characters • Format: "http[s]://<host>[:<port>][/[<path>]]" (quotation marks are not required but can be used) where: • <host>: Website address or IP address (IPv4/IPv6) • <port>: 1–65535 (Default: 80) • <path>: ASCII string • e.g., "http://abcd.net:80/1234" <state> (FOTA operation status) • 0—Not started • 1—Started (Continued on next page)

Table 9-2: Device Services Command Details (Continued)

Command	Description
+WDSTPF (continued)	**Device Services Third Party FOTA (continued)** **Example(s):** **AT+WDSTPF=?** +WDSTPF: 0,<addr> +WDSTPF: 1 OK **AT+WDSTPF?** +WDSTPF: 0,"http://abcd.net:80/1234" +WDSTPF: 1,0 OK // If the URL is incorrect, change it now **AT+WDSTPF=0,"http://efgh.net:80/1234"** OK **AT+WDSTPF=1** // Set start download OK +WDSI: 18,1 // Download 1% complete +WDSI: 18,5 // Download 5% complete +WDSI: 18,70 // Download 70% complete +WDSI: 18,100 // Download 100% complete +WDSI: 10 // Downloaded package stored in flash **AT+WDSR=4** // Activate OTA upgrade process OK +WDSI: 14 // OTA upgrade launched *[Wait for device to upgrade and reboot]* +WDSI: 16 // OTA upgrade finished successfully

10: Protocol Commands

10.1 Introduction

This chapter describes Internet Protocol (TCP, UDP, FTP, HTTP, SSL Certificate Manager) related commands.

Usage Notes

The following general usage notes apply to the AT commands described in this chapter:

- Session IDs—These protocol-specific AT commands share the same range of session IDs. Each session ID (<session_id> is a unique number in the range 1–12).
- IP address format—Unless otherwise specified, IP address parameters in the AT commands described in this chapter use the following formats:
  - IPv4—Dot-separated decimal (0–255) values using the format a1.a2.a3.a4
  - IPv6—Colon-separated hexadecimal (0–FFFF) values in the format a1:a2:a3:a4:a5:a6:a7:a8. Abbreviations are supported (e.g. 2001:0db8:3c4d:0015:0000:0000:1a2f:1a2b can be abbreviated as 2001:db8:3c4d:15::1a2f:1a2b)
- PDP context connection
  - A PDP connection starts when a session becomes active (e.g. using +KTCPNCX) and stops only if all sessions are closed or all sessions request to stop the connection.
    - By default, a PDP connection is requested to stop only when a session is closed by an Internet AT command (e.g. +KTCPCLOSE).
    - To configure the PDP connection deactivation behavior with respect to session errors, use +KIPOPT with <option_id>=3.

10.1.1 FTP Reply Codes

Table 10-1: FTP Reply Codes

Code	Description
110	Restart marker reply
120	Service ready in nnn minutes
125	Data connection already open: transfer starting
150	File status okay; about to open data connection
200	Command okay
202	Command not implemented, superfluous at this site
211	System status or system help reply
212	Directory status
213	File status
214	Help message
215	NAME system type
220	Service ready for new user
221	Service closing control connection. Logged out if appropriate. Unassigned (unallocated) number
225	Data connection open; no transfer in progress
226	Closing data connection. Requested file action successful (for example, file transfer or file abort)
227	Entering Passive Mode (<comma-separated IP address>,<comma-separated port>)
230	User logged in, proceed
250	Requested file action okay, completed
257	"PATHNAME" created
331	User name okay, need password
332	Need account for login
350	Requested file action pending further information
421	Service not available, closing control connection. This may be a reply to any command if the service knows it must shut down
425	Can't open data connection
426	Connection closed; transfer aborted
450	Requested file action not taken. File unavailable (e.g., file busy)
451	Requested action aborted: local error in processing
452	Requested action not taken. Insufficient storage space in system
500	Syntax error, command unrecognized. This may include errors such as command line too long

Table 10-1: FTP Reply Codes (Continued)

Code	Description
501	Syntax error in parameters or arguments
502	Command not implemented
503	Bad sequence of commands
504	Command not implemented for that parameter
530	Not logged in
532	Need account for storing files
550	Requested action not taken. File unavailable (e.g., file not found, no access)
551	Requested action aborted: page type unknown
552	Requested file action aborted. Exceeded storage allocation (for current directory or dataset)
553	Requested action not taken. File name not allowed

10.2 Command summary

Table 10-2 lists the commands described in this chapter:

Table 10-2: Protocol Commands

Command	Description	Page
+CGPADDR	Display module's PDP context addresses	118
+KCERTDELETE	Delete local certificate from the index	119
+KCERTSTORE	Store/display root CA and local certificates in internal storage	120
+KCNX_IND (notification)	Connection Status Notification—Unsolicited notification	122
+KCNXPROFILE	Query/Set default PDP context	123
+KFTPCFG	FTP configuration	124
+KFTPCFGDEL	Delete a configured FTP session	127
+KFTPCLOSE	Close the FTP connection	128
+KFTPCNX	Start the FTP connection	129
+KFTPDEL	Delete FTP files	131
+KFTP_IND	FTP status—Unsolicited notification	133
+KFTPLS	List the size of a specific file	134
+KFTPRCV	Receive FTP file	136
+KFTPSND	Send FTP file	139
+KHTTPCFG	Configure HTTP connection	142
+KHTTPCLOSE	Close HTTP connection	144
+KHTTPCNX	Start the HTTP connection	145
+KHTTPDEL	Delete configured HTTP session	146
+KHTTPGET	Get HTTP server information	147
+KHTTPHEAD	Get HTTP Headers	148
+KHTTPHEADER	Set the HTTP request header	149
+KHTTP_IND	HTTP status—Unsolicited notification	150
+KHTTPPOST	Send data to HTTP server	151
+KMQTTCFG	Configure MQTT client/broker messaging protocol parameters	153
+KMQTTCLOSE	Close connection to MQTT broker	155
+KMQTTCNX	Connect to MQTT broker	156
+KMQTT_DATA (notification)	MQTT data received—Unsolicited notification	157
+KMQTTDEL	Delete MQTT client session	158

Table 10-2: Protocol Commands

10.3 Command reference

Table 10-3: Protocol Command Details

Command	Description
+CGPADDR	**Display module's PDP context addresses** Display the module's address for a specific PDP context, or all contexts. **Password required:** No **Notes:** • Command can be used after +KTCPCNX, +KUDPCFG, etc., to display the local IP address of the module. • For IPv6, more than one PDP address corresponding to the interface may be displayed. **Usage:** • Execution (one context): **AT+CGPADDR=<cnx_cnf>** Response: +CGPADDR: <cnx_cnf>, <PDP_addr$_1$> OK Purpose: Display the address for the specified context. • Execution (all contexts): **AT+CGPADDR** Response: +CGPADDR: <cnx_cnf$_1$>, <PDP_addr$_1$> [+CGPADDR: <cnx_cnf$_2$>, <PDP_addr$_2$>] [...] OK Purpose: Display the addresses for all contexts. • Query List: **AT+CGPADDR=?** Purpose: Display valid parameter values. **Parameters:** <cnx_cnf> (PDP context configuration) • Specifies a particular PDP context configuration that has been defined by +CGDCONT. (e.g. <cnx_cnv=3> corresponds to CID=3 in +CGDCONT and +CGACT) <PDP_addr> (IP Address of module in PDP address space) • ASCII string

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KCERTDELETE	**Delete local certificate from the index** Enables choosing which certificate to delete from the index. **Password required:** No **Usage:** • Execution: **AT+KCERTDELETE=<data_type>[,<index>]** Response: OK *or* +CME ERROR: <err> Purpose: Select which local certificate to delete. • Query: **AT+KCERTDELETE?** Response: +KCERTDELETE: OK *or* +CME ERROR: <err> Purpose: Delete certificate. • Query List: **AT+KCERTDELETE=?** Purpose: Display valid execution format and parameter values. **Parameters:** <data_type> (Certificate type) • Valid range: • 0—Root certificate • 1—Local certificate <index> (Stored local certificate index) • Valid range: • 0—3: If <data_type> = 0 • 0—2: If <data_type> = 1

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KCERTSTORE	**Store/display root CA and local certificates in internal storage** Store Root Certificate Authority (CA) or local certificates in internal storage, or display all stored certificates. **Notes:** • Before using this command, it is highly recommended to configure the module for hardware flow control using AT+IFC. • The <index> parameter is the link between a local certificate and a private key (refer to +KPRIVKSTORE and +KCERTDELETE for more information). • The data session can end in the following ways: • Automatically, when <NbData> data bytes are sent or received, and the module returns to command state and returns OK. • Automatically, 10 seconds after "CONNECT" is received (i.e., the host most finish sending. • Additional methods to be supported in a future release: • By the host sending the end of file pattern "--EOF--Pattern--". • By the host sending "+++". • ATO is not available for this command. **Password required:** Yes **Usage:** • Execution: **AT+KCERTSTORE=<data_type>[,<NbData>[,<index>]]** Response: CONNECT *<File_data>* ← *Host sends data to store* OK *or* CONNECT *<File_data>* ← *Host sends data to store* +++ *or* *<EOF_pattern>* ← *Host wants to end the data session* +CME ERROR: <err> Purpose: Select which local certificate and how much bytes to store. • Query: **AT+KCERTSTORE?** Response: CONNECT root_cert,<index>,<NbData> ← *Root certificate with index=0* <File_data> [...] ← *Root certificates with index=1–3* local_cert,<index>,<NbData> ← *Local certificate with index=0* <File_data> [...] ← *Local certificate with index=1–2* OK *or* +CME ERROR: <err> Purpose: Checks which certificate to store. • Query List: **AT+KCERTSTORE=?** Purpose: Display valid execution format and parameter values. (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KCERTSTORE (continued)	**Store root CA and local certificates to internal storage (continued)** **Parameters:** <data_type> (Certificate type) • Valid range: • 0—Root certificate • 1—Local certificate <NbData> (Amount of bytes to be read or written) • Valid range: 1—4096 <index> (Stored local certificate index) • Valid range: • 0—3: If <data_type> = 0 • 0—2: If <data_type> = 1 <File_data> (File data in bytes) • String <EOF_pattern> (End of file pattern) • Note—This will be supported in a future release. • String • Valid value: "--EOF--Pattern--" **Example(s):** **AT+KCERTSTORE=0** ← *Store a Root CA certificate* CONNECT -----BEGIN CERTIFICATE----- CERTIFICATE1234567890abcdefghijklmnopqrstuvwxyz -----END CERTIFICATE----- --EOF--Pattern-- OK **AT+KCERTSTORE=1** ← *Store a local certificate* CONNECT -----BEGIN CERTIFICATE----- CERTIFICATEabcdefghijklmnopqrstuvwxyz1234567890 -----END CERTIFICATE----- --EOF--Pattern-- OK **AT+KCERTSTORE?** ← *Display all stored certificates* CONNECT root_cert,0,47 -----BEGIN CERTIFICATE----- CERTIFICATE1234567890abcdefghijklmnopqrstuvwxyz -----END CERTIFICATE----- root_cert,1,0 root_cert,2,0 root_cert,3,0 local_cert,0,47 -----BEGIN CERTIFICATE----- CERTIFICATEabcdefghijklmnopqrstuvwxyz1234567890 -----END CERTIFICATE----- local_cert,1,0 local_cert,2,0 OK

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KCNX_IND (notification)	**Connection Status Notification—Unsolicited notification** Unsolicited notification indicating the status of a connection attempt. To enable this notification, use +KURCCFG on page 190. **Notification format:** • For <status> = 0 or 1: **+KCNX_IND: <cnx_cnf>,<status>,<af>** • For <status> = 2: **+KCNX_IND: <cnx_cnf>,<status>,<attempt>,<nbtrial>,<tim1>** • For <status> = 3 or 6: **+KCNX_IND: <cnx_cnf>,<status>** • For <status> = 4: **+KCNX_IND: <cnx_cnf>,<status>,<attempt>** • For <status> = 5: **+KCNX_IND: <cnx_cnf>,<status>,<idletime>** **Parameters:** <cnx_cnf> (PDP context configuration) • Specifies a particular PDP context configuration that has been defined by +CGDCONT. (e.g., <cnx_cnv=3> corresponds to CID=3 in +CGDCONT and +CGACT) <status> (PDP connection status) • 0—Disconnected due to network • 1—Connected • 2—Failed to connect. <tim1> timer is started if <attempt> is less than <nbtrial>. • 3—Closed • 4—Connecting • 5—Idle time down counting started for disconnection • 6—Idle time down counting canceled <af> (IP address family type used for connection, compliant up to 3GPP Release 7) • Valid values: • 0—IPv4 • 1—IPv6 <tim1> (PDP activation reattempt timer, in seconds) • Integer • Valid range: 1–120 <attempt> (PDP connection attempt number) • Integer <nbtrial> (Max number of PDP activation attempts) • Integer • Valid range: 1–4 <idletime> (Max idle time, in seconds) • Integer • Valid range: 0–1800

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KCNXPROFILE	**Query/Set default PDP context** Display or set the default PDP profile that will be used by +KTCPCFG and +KUDPCFG if those commands do not specify a context. **Password required:** No **Usage:** • Execution: **AT+KCNXPROFILE=<cnx_cnf>** Response: OK Purpose: Set the current profile. • Query: **AT+KCNXPROFILE?** Response: +KCNXPROFILE: <cnx_cnf> OK Purpose: Display the current profile. • Query List: **AT+KCNXPROFILE=?** Purpose: Display valid execution format and parameter values. **Parameters:** <cnx_cnf> (PDP context configuration) • Specifies a particular PDP context configuration that has been defined by +CGDCONT. (e.g. <cnx_cnv=3> corresponds to CID=3 in +CGDCONT and +CGACT)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPCFG	**FTP configuration** Set or display the FTP configuration **Notes:** • The connection timeout for TCP socket is about 9 seconds with 3 retransmissions with a 3-second delay. • The default timeout for FTP is 30 seconds. • The maximum number of FTP connections is limited to 6. **Password required:** Yes **Usage:** • Execution: **AT+KFTPCFG=[<cnx cnf>],<server_name>[,<login>[,<password>[,<port_number>[,<mode>][,<start>][,<af>[,<secure>][,<cipher_index>]]]]]** Response: +KFTPCFG:<session_id> OK *or* +KFTP_ERROR: <session_id>,<ftp cause> Purpose: Configure an FTP session by setting the server name, login, password, port number and mode for FTP operations. • Query: **AT+KFTPCFG?** Response: +KFTPCFG: <session_id>,<cnx cnf>,<server_name>,<login>,<password>,<port_number>,<mode>,<started>,<af>,<secure>,<cipher_index> Purpose: Display the FTP session configuration. • Query List: **AT+KFTPCFG=?** Purpose: Display valid execution format and parameter values. **Parameters:** <cnx cnf> (Index of a set of parameters) • Valid range: 1 • This command is used for configuring one FTP session. <session_id> (FTP session index) • Integer • Valid range: 1–12 <server_name> (FTP server IP address string or server's domain name) • String <login> (User name to be used during the FTP connection) • String • String length: 0–64 bytes <password> (Password to be used during the FTP connection) • String • String length: 0–64 bytes <port_number> (Remote command port) • Integer • Valid range: 1–65535 • Default: 21 (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPCFG (continued)	**FTP configuration (continued)** <mode> (FTP connection initiator) • Integer • Valid value: • 1—Passive. Indicates that the client is the initiator of the FTP data connection to avoid the proxy filtrate. Note: The passive data transfer process listens on the data port for a connection from the active transfer process in order to open the data connection. Only passive mode is currently supported. Active mode is internally switched to passive. <start> (FTP connection start method) • 0—Start the FTP connection using +KFTPCNX • 1—Start the FTP connection immediately <started> (FTP connection start state) • 0—FTP connection not yet started • 1—FTP connection was started <af> (IP address family type used for the connection) • 0—IPv4 • 1—IPv6 <secure> (Explicit FTPS enabled) • 0—Unsecured • 1—Secured via TLS <cipher_index> (Cipher suite profile index to use for a secured socket) • Defined by +KSSLCRYPTO • Valid range: 0–7 <ftp_cause> (Cause of FTP connection failure) • Integer • 0—Sending or retrieving failed due to request timeout • 1—Failed to connect to the server due to DNS resolution failure • 2—FTP connection error due to internal trouble • 3—Failed to download due to connection timeout • 4—No network available • 5—Flash access error • 6—Flash memory full • Note: For XXX or three-digit reply codes from the FTP server, see Table 10-1, FTP Reply Codes. (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPCFG (continued)	**FTP configuration (continued)** **Example(s):** • Command: AT+KFTPCFG=1,"ftp.connect.com","username","password",21,1 Response: +KFTPCFG: 1 OK • Command: AT+KFTPCFG=? Response: +KFTPCFG: (1),<server_name>,(0-64),(0-64),(1-65535),(0-1),(0-1),(0-1) OK • Command: AT+KFTPCFG? Response: +KFTPCFG: 1,1,"ftp.connect.com","username","",21,1,0,0 OK

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPCFGDEL	**Delete a configured FTP session** Enables deleting a specific FTP session that was previously configured. **Notes:** • The session must be closed (using +KFTPCLOSE) before using this command **Password required:** No **Usage:** • Execution: **AT+KFTPCFGDEL=<session_id>** Response: OK *or* +CME ERROR: <err> Purpose: Delete a specific FTP session. • Query List: **AT+KFTPCFGDEL=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (FTP session index) • Integer • Valid range: 1–12 **Example(s):** • Command: AT+KFTPCFGDEL=? Response: +KFTPDEL: (1-12) OK • Command: AT+KFTPCFGDEL=1 Response: OK

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPCLOSE	**Close the FTP connection** Closes the FTP server connection. **Password required:** No **Usage:** • Execution: **AT+KFTPCLOSE=<session_id>[,<keep_cfg>]** Response: OK Purpose: Close a specific FTP session. • Query List: **AT+KFTPCLOSE=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (FTP session index) • Integer • Valid range: 1–12 <status> (Specifies whether to delete the session configuration after closing it) • Integer • Valid range: • 0—Delete the session configuration • 1—Keep the session configuration **Example(s):** • Command: AT+KFTPCLOSE=? Response: +KFTPCLOSE: (1-12),(0-1) OK • Command: AT+KFTPCLOSE=1,1 Response: OK

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPCNX	**Start the FTP connection** Enables connection to the FTP. **Notes:** • This command is used to start the FTP connection, created by +KFTPCFG, with <start>=0. • The result of the FTP connection is indicated by URC. **Password required:** No **Usage:** • Execution: **AT+KFTPCNX=<session_id>** Response: OK *or* NO CARRIER *or* +CME ERROR: <err> *or* +KFTP_ERROR: <session_id>,<ftp cause> Purpose: Configure a connection and receive a session ID. • Query: **AT+KFTPCNX?** Response: OK Purpose: Display the configurations for the sessions. • Query List: **AT+KFTPCNX=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (FTP session index) • Integer • Valid range: 1–12 <ftp_cause> (Cause of FTP connection failure) • Integer • 0—Sending or retrieving failed due to request timeout • 1—Failed to connect to the server due to DNS resolution failure • 2—FTP connection error due to internal trouble • 3—Failed to download due to connection timeout • 4—No network available • 5—Flash access error • 6—Flash memory full • Note: For XXX or three-digit reply codes from the FTP server, see Table 10-1, FTP Reply Codes. (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPCNX (continued)	**Start the FTP connection (continued)** **Example(s):** • Command: AT+KFTPCNX=? Response: +KFTPCNX: (1-12) OK • Command: AT+KFTPCNX=1 Responses: ERROR +KFTP_ERROR: 1,2 *or* +KCNX_IND: 1,1,0 +KFTP_IND: 1,1

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPDEL	**Delete FTP files** Enables deleting FTP files. **Notes:** • Before using this command, you must have an FTP connection via AT+KFTPCFG. • Results are indicated via URC. **Password required:** No **Usage:** • Execution: **AT+KFTPDEL=<session_id>,[<server_path>],<file_name>[,<type>]** Response: OK *or* +CME ERROR <err> NO CARRIER +KFTP_ERROR: <session_id>,<ftp cause> Purpose: Configure a connection and receive a session ID and error notice if applicable. • Query: **AT+KFTPDEL?** Response: ERROR Purpose: Produces an error response. • Query List: **AT+KFTPDEL=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (FTP session index) • Integer • Valid range: 1–12 <server_path> (Indicates the file path to be deleted) • String • An empty string or no string indicates that the upload is done from the path given by the FTP server. <file_name> (Indicates the file name to be deleted) • String <type> (Indicates the file type, ASCII or binary, to be transferred) • Numeric type • Valid values: • 0–Binary (default value) • 1–ASCII (not supported) (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPDEL (continued)	**Delete FTP files (continued)** <ftp_cause> (Indicates the cause of the FTP connection failure) • Integer • Valid values: • 0–Sending or the retrieving was impossible due to request timeout • 1–Cannot connect to the server due to DNS resolution failure • 2–Unable to download a file due to connection issues • 3–Unable to download due to connection timeout • 4–No network available • 5–Flash access trouble • 6–Flash memory full • Note: XXX three digits, reply codes from FTP server. See FTP Reply Codes. **Example(s):** • Command: AT+KFTPDEL=? Response: +KFTPDEL: (1-12),<server_path>,<file_name>,(0) OK

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTP_IND (notification)	**FTP status—Unsolicited notification** Unsolicited notification that indicates the FTP's status. **Usage:** • Notification: **+KFTP_IND: <session_id>,<status>[,<data_len>]** Purpose: Indicate the FTP status together with the session ID and the data's byte length. **Parameters:** <session_id> (FTP session index) • Integer • Valid range: 1–12 <status> (Indicates the status of the FTP session) • Integer • Valid range: • 1–Session is set up and ready for operation • 2–The last FTP command has been executed successfully <data_len> (Data byte length) • Byte length of data downloaded or uploaded to and from the terminal via +KFTPRCV or +KFTPSND.

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPLS	**List the size of a specific file** **Notes:** • Before using this command, you must have an FTP connection via AT+KFTPCFG. • Results are indicated via URC **Password required:** No **Usage:** • Execution: **AT+KFTPLS=<session_id>,[<server_path>],<file_name>[,<type>]** Response: OK *or* +CME ERROR <err> NO CARRIER +KFTP_ERROR: <session_id>,<ftp cause> Purpose: Show the size of a specific file name. • Query List: **AT+KFTPLS=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (FTP session index) • Integer • Valid range: 1–12 <server_path> (Indicates the file path to be deleted) • String • An empty string or no string indicates that the upload is done from the path given by the FTP server. <file_name> (Indicates the file name to be listed • String <type> (Indicates the file type, ASCII or binary, to be transferred) • Numeric type • Valid values: • 0–Binary (default value) • 1–ASCII (not supported) <ftp_cause> (Indicates the cause of the FTP connection failure) • Integer • Valid values: • 0–Sending or the retrieving was impossible due to request timeout • 1–Cannot connect to the server due to DNS resolution failure • 2–Unable to download a file due to connection issues • 3–Unable to download due to connection timeout • 4–No network available • 5–Flash access trouble • 6–Flash memory full • Note: XXX three digits, reply codes from FTP server. See FTP Reply Codes. (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPLS (continued)	**List the size of a specific file** **Example(s):** • Command: AT+KFTPLS=? Response: +KFTPLS: (1-12),<server_path>,<file_name>,(0) OK • Command: AT+KFTPLS=1,,"filename.txt" Response: +KFTPLS: filename.txt 24 OK

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPRCV	**Receive FTP file** Receive an FTP file from an FTP server. The received data is directed to the AT port (i.e., displayed in the terminal window); the data is not stored to the module's internal memory. **Notes:** • Before using this command, an FTP connection must have been achieved using +KFTPCFG. • After sending the +KFTPRCV command, the user will receive the entire data stream. • The data session can end in the following ways: • Automatically, when the entire file has been received, and the module returns to command state and returns OK. • By the host sending "+++". • Additional methods to be supported in a future release: • By the host sending the end of file pattern "--EOF--Pattern--". The transfer ends and the host receives "NO CARRIER" • If the data session is ended before the file is fully received, the host can attempt to resume the download by repeating the execution command using the same <server_path> with the <offset> set to indicate where in the file to resume downloading. (e.g., If the file is 10 MB total but the download is ended after 3 MB has been received, the host can set the <offset>=3145728 (3 MB) to resume downloading the rest of the file.) However, if the FTP server does not support the resume feature, the host will receive a +KCTP_ERROR: 421, 500, 501, 502, or 530. See Table 10-1, FTP Reply Codes for details. **Password required:** No **Usage:** • Execution: **AT+KFTPRCV=<session_id>,[<local_uri>],[<server_path>],<file_name>[,<type_of_file>[,<offset>][,<size>]]** Response: CONNECT *<data>* *[<EOF_pattern>]* ← *End of file string (to be supported in a future release)* OK *or* +CME ERROR<err> *or* NO CARRIER *or* +KFTP_ERROR: <session_id>,<ftp cause> Purpose: Configure a connection and receive a session ID and error notice if applicable. • Query: **AT+KFTPRCV?** Response: ERROR Purpose: Produces an error response. • Query List: **AT+KFTPRCV=?** Purpose: Display valid execution format and parameter values. (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPRCV (continued)	**Receive FTP file (continued)** **Parameters:** <session_id> (FTP session index) • Integer • Valid range: 1–12 <local_uri> (Reserved for compatibility of command syntax) • String • This argument must be empty. <server_path> (Indicates the file path for the download) • String • An empty string or no string indicates that the download is done from the path given by the FTP server. <file_name> (Indicates the file name to be downloaded) • String <type_of_file> (Indicates the file type, ASCII or binary, to be transferred) • Numeric type • Valid values: • 0—Binary (default value) • 1—ASCII (not supported) <offset> (Indicates the offset to resume transfer) • Integer • Valid range: 0–4294967295 • Note: When downloading a file and transmitting to a serial link, the module will use the <offset> value and resume transfer from here. <size> (Indicates the size to resume transfer) • Integer • Valid range: 0–4294967295 • Note: When a downloading file and transmitting to a serial link, the module will use the <size> value to indicate the amount of bytes to be received. <EOF_pattern> (End of file pattern) • Note—This will be supported in a future release. (It does not appear as of the release date of this document.) • String • Valid value: "--EOF--Pattern--" <ftp_cause> (Indicates the cause of the FTP connection failure) • Integer • Valid values: • 0—Sending or the retrieving was impossible due to request timeout • 1—Cannot connect to the server due to DNS resolution failure • 2—FTP connection error due to internal trouble • 3—Unable to download due to connection timeout • 4—No network available • 5—Unable to access flash • 6—Flash memory full • Note: XXX three digits, reply codes from FTP server. See FTP Reply Codes. (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPRCV (continued)	**Receive FTP file (continued)** **Example(s):** • Command: AT+KFTPRCV? Response: ERROR • Command: AT+KFTPRCV=? Responses: +KFTPRCV: (1-12),<local_uri>,<server_path>,<file_name>,(0),(0-4294967295), (0-4294967295) OK • Command: AT+KFTPRCV=1,,,"filename.txt" Response: CONNECT …data… OK +KFTP_IND: 1,2,10

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPSND	**Send FTP file** Send an FTP file from the module to the FTP server. **Notes:** • Before using this command, you must have an FTP connection via AT+KFTPCFG. • After issuing the +KFTPSND=... command, the host must send the entire file in one data stream. • When the host has finished sending the file, the host must end the data session. The data session can end in the following ways: • By the host sending "+++". • Additional methods to be supported in a future release: • By the host sending the end of file pattern (<EOF_pattern>). • ATO is not available for this command. **Password required:** No **Usage:** • Execution: **AT+KFTPSND=<session_id>,[<local_uri>],[<server_path>],<file_name>[,<type_of_life>][,<append>][,<offset>][,<size>]** Response: CONNECT data … OK *or* CONNECT data … <EOF_pattern> +CME ERROR <err> *or* NO CARRIER *or* +KFTP_ERROR: <session_id>,<ftp cause> Purpose: Configure a connection and receive a session ID and error notice if applicable. • Query: **AT+KFTPSND?** Response: ERROR Purpose: Produces an error response. • Query List: **AT+KFTPSND=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (FTP session index) • Integer • Valid range: 1–12 <local_uri> (Reserved for compatibility of command syntax) • String • Note: This argument must be empty. <server_path> (Indicates the file path for the upload) • String • An empty string or no string indicates that the upload is done from the path given by the FTP server. (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPSND (continued)	**Send FTP file (continued)** <file_name> (Indicates the file name to be uploaded) • String <type_of_file> (Indicates the file type, ASCII or binary, to be transferred) • Numeric type • Valid values: • 0–Binary (default value) • 1–ASCII (not supported) <append> (Indicates when to use append for uploading) • Numeric type • Valid values: • 0—Do not use append. (default value). If the file already exists then the file will be overridden • 1—Use append. If the file already exists then the data will be appended at the end of the file. Otherwise, the file will be created. <offset> (Indicates the offset to resume transfer) • Integer • Valid range: 0–4294967295 • Note: When sending a file and transmitting to a serial link, the module will use the <offset> value and resume transfer from here. <size> (Indicates the size to resume transfer) • Integer • Valid range: 0–4294967295 • Note: When sending a file and transmitting to a serial link, the module will use the <size> value to indicate the amount of bytes to send. <EOF_pattern> (End of file pattern) • Note—This will be supported in a future release. (It is not supported as of the release date of this document.) • String • Valid value: "--EOF--Pattern--" <ftp_cause> (Indicates the cause of the FTP connection failure) • Integer • Valid values: • 0–Sending or the retrieving was impossible due to request timeout • 1–Cannot connect to the server due to DNS resolution failure • 2–Unable to download a file due to connection issues • 3–Unable to download due to connection timeout • 4–No network available • 5–Unable to access flash • 6–Flash memory full • Note: XXX three digits, reply codes from FTP server. See FTP Reply Codes. (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KFTPSND (continued)	**Send FTP file (continued)** **Example(s):** • Command: AT+KFTPSND=? Response: +KFTPSND: (1-12),<local_uri>,<server_path>,<file_name>,(0),(0-1), (0-4294967295),(0-4294967295) OK

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KHTTPCFG	**Configure HTTP connection** Set or display HTTP connection configuration. **Notes:** • <http_port> and <http_server> define the port and the IP address of the remote server one wants to connect • For <af> = 1 (IPV6), server address (<http_server>) in IP address string format can be optionally quoted with square brackets "[]", e.g. [FEDC:BA98:7654:3210:FEDC:BA98:7654:3210] **Password required:** No **Usage:** • Execution: **AT+KHTTPCFG=<cnx_cnf>, <server-name/ip>[, <http_port>[, <http_version>[,<login>[, <password>[, <start>[, <af>[, <cipher_index>]]]]]]]** Response: +KHTTPCFG: <session_id> OK *or* +CME ERROR: <err> Purpose: Configure a connection and receive an HTTP session ID. • Query: **AT+KHTTPCFG?** Response: +KHTTPCFG: <session_id>, <cnx_cnf>, <server-name/ip>, <http_port>, <http_version>, <login>, <password>, <started>, <af>, <cipher_index> OK Purpose: Display the configurations for all HTTP sessions. • Query List: **AT+KHTTPCFG=?** Purpose: Display valid execution format and parameter values. **Parameters:** <cnx_cnf> (PDP context configuration) • Valid value: 1 <session_id> (HTTP session index) • Integer <server-name/ip> (IP address string or explicit name of the remote server) • String <http_port> (Port on remote server) • Valid range: 1–65535 • 80—HTTP 1.1 (Default) • 443—HTTP 1.1 over TLS (HTTPS) <http_version> (HTTP version) • 0—HTTP 1.1 (Default) • 2—HTTP 1.1 over TLS (HTTPS) <login> (User name to be used during the HTTP connection) • String <password> (Password to be used during the HTTP connection) • String (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KHTTPCFG (continued)	**Configure HTTP connection (continued)** <start> (When to start the HTTP connection) • 0—Start the HTTP connection later using +KHTTPCNX • 1—Start the HTTP connection immediately <started> (HTTP connectdion start status) • 0—Connection has not started yet • 1—Connection has started <af> (IP address family type used for connection) • Valid values: • 0—IPv4 (Default) • 1—IPv6 <cipher_suite> (Cipher suite profile index to use for a secured socket) • Integer value • Defined by +KSSLCRYPTO

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KHTTPCLOSE	**Close HTTP connection** Close an HTTP session. **Password required:** No **Usage:** • Execution: **AT+KHTTPCLOSE=<session_id>[,<keep_cfg>]** Response: OK *or* CME ERROR: <err> Purpose: Close the specified HTTP connection. • Query List: **AT+KHTTPCLOSE=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (HTTP session index) • Integer <keep_cfg> (Delete/keep session configuration after closing) • 0—Delete the session configuration • 1—Keep the session configuration

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KHTTPCNX	**Start the HTTP connection** **Notes:** • This command is used to start the HTTP connection created by +KHTTPCFG with <start>=0. • +KHTTPGET automatically starts the connection if it has not been started before using AT+KHTTPCNX. **Usage:** • Execution: **AT+KHTTPCNX=<session_id>** Response: OK +KHTTP_IND: <session_id>,<status>[,<data_len>,<st_code>,<st_reason>] *or* +KHTTP_ERROR: <session_id>,<http_notif> *or* +CME ERROR: <err> Purpose: Start the specified HTTP connection. • Query List: **AT+KHTTPCNX=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (HTTP session index) • Integer <http_notif> (Cause of HTTP connection failure) • Integer • Valid values: • 4—DNS error • 5—HTTP connection error due to internal trouble • 6—HTTP connection timeout • 7—Flash access trouble • 8—Flash memory full • 9—Triple-plus (+++) error (switch to command mode) • 10—HTTP got no data • 11—HTTP got partial data

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KHTTPDEL	**Delete configured HTTP session** **Notes:** • Session must be closed using +KHTTPCLOSE before using this command. **Password required:** No **Usage:** • Execution: **AT+KHTTPDEL=<session_id>** Response: OK *or* +CME ERROR: <err> Purpose: Delete the specified HTTP connection. • Query List: **AT+KHTTPDEL=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (HTTP session index) • Integer

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KHTTPGET	**Get HTTP server information** Get the server information for an HTTP session. **Notes:** • The data session can end in the following ways: • By the host sending "+++". • Additional methods to be supported in a future release: • By the host sending the end of file pattern "--EOF--Pattern--". **Password required:** No **Usage:** • Execution: **AT+KHTTPGET=<session_id>[,<request_uri>[, <show_resp>]]** Response: CONNECT <*data*> <*EOF_pattern*> ← *End of file string (to be supported in a future release)* OK +KHTTP_IND: <session_id>,<status>[,<data_len>, <st_code>,<st_reason>] *or* NO CARRIER +KHTTP_ERROR: <session_id>,<http_notif> *or* +CME ERROR: <err> Purpose: Get server information for the specified HTTP session. • Query List: **AT+KHTTPGET=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (HTTP session index) • Integer <request_uri> (Information URL to get during the HTTP connection) • String <http_notif> (Cause of HTTP connection failure) • 4—DNS error • 5—HTTP connection error due to internal trouble • 6—HTTP connection timeout • 7—Flash access trouble • 8—Flash memory full • 9—Triple-plus (+++) error (switch to command mode) • 10—HTTP got no data • 11—HTTP got partial data <show_resp> (Show or hide HTTP response and HTTP headers) • 0—Do not show response and headers • 1—Show response and headers (Default) <EOF_pattern> (End of file pattern) • Note—This will be supported in a future release. (It does not appear as of the release date of this document.) • String • Valid value: "--EOF--Pattern--"

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KHTTPHEAD	**Get HTTP Headers** **Notes:** • This method is identical to GET except that the server MUST NOT return a message-body in the response. The meta-information contained in the HTTP headers in response to a HEAD request SHOULD be identical to the information sent in response to a GET request. • The data session can end in the following ways: • By the host sending "+++". • Additional methods to be supported in a future release: • By the host sending the end of file pattern "--EOF--Pattern--". • HTTP does not support ATO. **Usage:** • Execution: **AT+KHTTPHEAD=<session_id>,<request_uri>** Response: CONNECT …<EOF_pattern> ← *End of file string (to be supported in a future release)* OK +KHTTP_IND: <session_id>,<status>[,<data_len>, <st_code>,<st_reason>] *or* NO CARRIER +KHTTP_ERROR: <session_id>,<http_notif> *or* +CME ERROR: <err> Purpose: Request HTTP headers from the server. • Query List: **AT+KHTTPHEAD=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (HTTP session index) • Integer <request_uri> (Information URL to get during the HTTP connection) • String <EOF_pattern> (End of file pattern) • Note—This will be supported in a future release. (It does not appear as of the release date of this document.) • String • Valid value: "--EOF--Pattern--" <http_notif> (Cause of HTTP connection failure) • Integer • Refer to +KHTTPGET

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KHTTPHEADER	**Set the HTTP request header** **Notes:** • The data session can end in the following ways: • (To be supported in a future release) The host must send the end of file pattern "--EOF--Pattern--" to finish sending data, then the module will return to command mode. • Automatically, if there is no response from the server for 20 seconds. (A CME_ERROR will be received.) **Usage:** • Execution: **AT+KHTTPHEADER=<session_id>** Response: CONNECT OK *or* NO CARRIER +KHTTP_ERROR: <session_id>,<http_notif> *or* +CME ERROR: <err> Purpose: Request HTTP headers from the server. • Query: **AT+KHTTPHEADER?** Response: +KHTTPHEADER: <session_id>,<count> […] OK Purpose: Display valid execution format and parameter values. • Query List: **AT+KHTTPHEADER=?** Response: +KHTTPHEADER: (list of possible <session_id>s) OK Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (HTTP session index) • Integer <count> (Count of HTTP headers) • Integer

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KHTTP_IND (notification)	**HTTP status—Unsolicited notification** Unsolicited notification indicating the HTTP's status. **Notification format:** **+KHTTP_IND: <session_id>,<status>[,<data_len>, <st_code>, <st_reason>]** **Parameters:** <session_id> (HTTP session index) • Integer <status> (Indicates the status of the HTTP session) • Integer • Valid range: • 0—Session is disconnected • 1—Session is set up and ready for operation • 3—The last HTTP command has been executed successfully <data_len> (Data byte length) • Byte length of data downloaded or uploaded to and from the terminal (using +KHTTPHEAD, +KHTTPGET or +KHTTPPOST). <st_code> (HTTP response status code) • Integer <st_reason> (HTTP response status reason string • String

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KHTTPPOST	**Send data to HTTP server** **Notes:** • The data session can end in the following ways: • By the host sending "+++". • By the host sending the end of file pattern "--EOF--Pattern--". • ATO is not available for this command. **Usage:** • Execution: AT+KHTTPPOST=<session_id>,<local_uri>,<request_uri>[,<show_resp>] Response: CONNECT …<EOF_pattern> ← *End of file string* OK +KHTTP_IND: <session_id>,<status>[,<data_len>,<st_code>,<st_reason>] *or* NO CARRIER +KHTTP_ERROR: <session_id>,<http_notif> *or* +CME ERROR: <err> Purpose: Send data to HTTP server. • Query List: **AT+KHTTPPOST=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (HTTP session index) • Integer <local_uri> (Reserved for compatibility of command syntax) • String • This argument must be empty. <request_uri> (Request data of the HTTP connection) • String <http_notif> (Cause of HTTP connection failure) • Integer • Valid values: • 4—DNS error • 5—HTTP connection error due to internal trouble • 6—HTTP connection timeout • 7—Flash access trouble • 8—Flash memory full • 9—Triple-plus (+++) error (switch to command mode) • 10—HTTP got no data • 11—HTTP got partial data <show_resp> (Shows or hides HTTP headers) • Valid values: • 0—Do not show HTTP headers, show HTTP body only • 1—Show HTTP headers and Body (Default) (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
KHTTPPOST (continued)	**Send data to HTTP server (continued)** <EOF_pattern> (End of file pattern) • Note—This will be supported in a future release. (It does not appear as of the release date of this document.) • String • Valid value: "--EOF--Pattern--"

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KMQTTCFG	**Configure MQTT client/broker messaging protocol parameters** Configure MQTT client and broker parameters. **Password required:** No **Usage:** • Execution: **AT+KMQTTCFG=<secure>,<server>,<port>,<version>,<client_id>[,[<keepalive_interval>],[<clean_session>],<will_flag>,<topic_name>,<message>,<retained>,<qos>[,[<username>],[<password>]],[<cipher_index>],[<alpn_list>]]** Response: +KMQTTCFG: <socket_ID> OK Purpose: Configure the MQTT client parameters. (Note—The command always returns 0 for the <socket_ID>.) • Query: **AT+KMQTTCFG?** Response: +KMQTTCFG: <socket_ID>,<secure>,<server>,<port>,<version>,<client_id>,<keepalive_interval>,<clean_session>,<will_flag>,<topic_name>,<message>,<retained>,<qos>,<username>,<password>,<cipher_index>,<alpn_list> OK Purpose: Display the currently configured client parameters. • Query List: **AT+KMQTTCFG=?** Purpose: Display valid execution format and parameter values. **Parameters:** <secure> (MQTT connection security method) • 0—No security • 1—TLS (Transport Layer) <server> (MQTT broker) • ASCII string (quotation marks are not required but can be used) • e.g., broker.hivemq.com <port> (MQTT broker port) • Valid port number range: 0–65535 <version> (MQTT version) • 3—MQTT version 3.1 • 4—MQTT version 3.1.1 (Default) <client_id> (MQTT client ID) • ASCII string (quotation marks are not required but can be used) • e.g., "QCX216" <keepalive_interval> (Maximum time allowed between MQTT packets sent by the client, in seconds) • Valid range: 0–65535 • 0—Disable • 120—Default <clean_session> (Clean Session flag) • 0—Client requests a clean session from the MQTT broker • 1—Client requests a persistent session from the MQTT broker (Default) (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KMQTTCFG (continued)	**Configure MQTT client/broker messaging protocol parameters (continued)** <will_flag> (Will Session flag) • 0—Disable the Last Will and Testament (LWT) (Default) • 1—Enable the Last Will and Testament <topic_name> (Last Will and Testament topic name) • ASCII string (quotation marks are not required but can be used) • e.g., "home/LWTMessage" <message> (Last Will and Testament message) • ASCII string (quotation marks are not required but can be used) • e.g., "qcx216 offline" <retained> (Last Will and Testament Retained flag) • 0—Not retained (Default) • 1—Retained <qos> (Last Will and Testament QOS) • 0—At most once • 1—At least once • 2—Exactly once <username> (MQTT username for broker authentication) • ASCII string (quotation marks are not required but can be used) <password> (MQTT password for broker authentication) • ASCII string (quotation marks are not required but can be used) <cipher_index> (Cipher suite profile index to use for a secured socket) • Value is defined by +KSSLCRYPTO <alpn_list> (ALPN protocol name list for MQTT broker authentication) • ASCII string (quotation marks are not required but can be used) • The list supports one ALPN name. <socket_ID> (Unique MQTT client socket ID) • Unique integer value identifying the MQTT client to the MQTT broker • Valid values: 0 Note—Only one socket is supported, and the ID is always 0. **Example(s):** **AT+KMQTTCFG=0,broker.hivemq.com,1883,4,"QCX216",120,1,1,"home/LWTMessage","qcx216 offline",1,0,,,,""** +KMQTTCFG: 0 OK **AT+KMQTTCFG?** +KMQTTCFG: 0,0,"broker.hivemq.com",1883,4,"QCX216",120,1,1,"home/LWTMessage","qcx216 offline",1,0,"","",0,"" OK

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KMQTTCLOSE	**Close connection to MQTT broker** Close the connection to the MQTT broker. **Password required:** No **Notes:** • This command does not delete the session configuration. **Usage:** • Execution: **AT+KMQTTCLOSE=<socket_ID>** Response: OK *or* +CME_ERROR: <cme_err> Purpose: Close the connection to the MQTT broker. **Parameters:** <socket_ID> (MQTT client socket ID) • Unique integer value assigned by +KMQTTCFG, which identifies the MQTT client to the broker <cme_err> (CME error code) • 910—CME_ER_BAD_SESSION_ID • 916—CME_ER_PARAMETER_INVALID_RANGE

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KMQTTCNX	**Connect to MQTT broker** Establish a connection to the MQTT broker. **Password required:** No **Usage:** • Execution: **AT+KMQTTCNX=<socket_ID>** Response: OK +KMQTT_IND: <socket_ID>,<status> ... *or* +CME_ERROR: <cme_err> Purpose: Connect to the MQTT broker. **Parameters:** <socket_ID> (MQTT client socket ID) • Unique integer value assigned by +KMQTTCFG, which identifies the MQTT client to the broker <status> (MQTT connection status) • 0—Connection aborted error • 1—Connection successful (CONNACK received from the MQTT broker) • 2—Subscribed to a topic successful (SUBACK received from the MQTT broker) • 3—Unsubscribed to a topic successful (UNSUBACK received from the MQTT broker) • 4—Message published successful (PUBACK received from the MQTT broker) • 5—Generic error • 6—Socket open successful <cme_err> (CME error code) • 910—CME_ER_BAD_SESSION_ID • 916—CME_ER_PARAMETER_INVALID_RANGE • 922—CME_ER_SESSION_INVALID_STATE

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KMQTT_DATA (notification)	**MQTT data received—Unsolicited notification** Unsolicited notification indicating MQTT data is incoming. **Notification format:** **+KMQTT_DATA: <socket_ID>,<topic_name>,<payload>** **Parameters:** <socket_ID> (MQTT client socket ID) • Unique integer value assigned by +KMQTTCFG, which identifies the MQTT client to the broker <topic_name> (Name of topic in MQTT session) • ASCII string • Examples: "home/led", "home/yard/DHT11", "Sensor", etc. <payload> (Message payload) • ASCII string • Maximum length: 1024 bytes

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KMQTTDEL	**Delete MQTT client session** Delete the MQTT client session. **Password required:** No **Usage Requirements:** • Before using this command, the session must be closed using +KMQTTCLOSE. **Usage:** • Execution: **AT+KMQTTDEL=<socket_ID>** Response: OK *or* +CME_ERROR: <cme_err> Purpose: Delete the MQTT client session. **Parameters:** <socket_ID> (MQTT client socket ID) • Unique integer value assigned by +KMQTTCFG, which identifies the MQTT client to the broker <cme_err> (CME error code) • 910—CME_ER_BAD_SESSION_ID • 916—CME_ER_PARAMETER_INVALID_RANGE

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KMQTT_IND (notification)	**MQTT status received—Unsolicited notification** Unsolicited notification indicating MQTT status. **Notification format:** **+KMQTT_IND: <socket_ID>,<status>** **Parameters:** <socket_ID> (MQTT client socket ID) • Unique integer value assigned by +KMQTTCFG, which identifies the MQTT client to the broker <status> (MQTT connection status) • 0—Connection aborted error • 1—Connection successful (CONNACK received from the MQTT broker) • 2—Subscribed to a topic successful (SUBACK received from the MQTT broker) • 3—Unsubscribed to a topic successful (UNSUBACK received from the MQTT broker) • 4—Message published successful (PUBACK received from the MQTT broker) • 5—Generic error • 6—Socket open successful

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KMQTTPUB	**Publish message to MQTT client session** Publish a message to an MQTT session topic. **Password required:** No **Usage Requirements:** • Before using this command, the session must be connected to the MQTT broker using +KMQTTCNX. **Usage:** • Execution: **AT+KMQTTPUB=<socket_ID>,<topic_name>,<qos>,<retained>,<payload>** Response: OK +KMQTT_IND: <socket_ID>,<status> *or* +CME_ERROR: <cme_err> Purpose: Publish a message to an MQTT session topic. **Parameters:** <socket_ID> (MQTT client socket ID) • Unique integer value assigned by +KMQTTCFG, which identifies the MQTT client to the broker <topic_name> (Name of topic in MQTT session) • ASCII string <qos> (Last Will and Testament QOS) • 0—At most once • 1—At least once • 2—Exactly once <retained> (Retained flag configuration) • 0—Not retained • 1—Retained <payload> (Message payload) • ASCII string • Maximum length: 1024 bytes <status> (MQTT connection status) • 0—Connection aborted error • 1—Connection successful (CONNACK received from the MQTT broker) • 2—Subscribed to a topic successful (SUBACK received from the MQTT broker) • 3—Unsubscribed to a topic successful (UNSUBACK received from the MQTT broker) • 4—Message published successful (PUBACK received from the MQTT broker) • 5—Generic error • 6—Socket open successful <cme_err> (CME error code) • 910—CME_ER_BAD_SESSION_ID • 916—CME_ER_PARAMETER_INVALID_RANGE

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KMQTTSUB	**Subscribe to MQTT session topic** Submit a request to subscribe to a specific MQTT topic. **Password required:** No **Usage Requirements:** • Before using this command, the session must be connected to the MQTT broker using +KMQTTCNX. **Notes:** • Any incoming messages to the subscribed topic will be shown as a +KMQTT_DATA URC in the format: +KMQTT_DATA: <socket_ID>,"<topic_name>","<payload>" **Usage:** • Execution: **AT+KMQTTSUB=<socket_ID>,<topic_name>,<qos>** Response: OK +KMQTT_IND: <socket_ID>,<status> *or* +CME_ERROR: <cme_err> Purpose: Subscribe to a specific MQTT session topic. **Parameters:** <socket_ID> (MQTT client socket ID) • Unique integer value assigned by +KMQTTCFG, which identifies the MQTT client to the broker <topic_name> (Name of MQTT session topic to subscribe to) • ASCII string <qos> (Last Will and Testament QOS) • 0—At most once • 1—At least once • 2—Exactly once <status> (MQTT connection status) • 0—Connection aborted error • 1—Connection successful (CONNACK received from the MQTT broker) • 2—Subscribed to a topic successful (SUBACK received from the MQTT broker) • 3—Unsubscribed to a topic successful (UNSUBACK received from the MQTT broker) • 4—Message published successful (PUBACK received from the MQTT broker) • 5—Generic error • 6—Socket open successful <cme_err> (CME error code) • 910—CME_ER_BAD_SESSION_ID • 916—CME_ER_PARAMETER_INVALID_RANGE

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KMQTTUNSUB	**Unsubscribe from MQTT session topic** Submit a request to unsubscribe from a specific MQTT topic. **Password required:** No **Usage Requirements:** • Before using this command, the session must be connected to the MQTT broker using +KMQTTCNX. **Usage:** • Execution: **AT+KMQTTUNSUB=<socket_ID>,<topic_name>** Response: OK +KMQTT_IND: <socket_ID>,<status> *or* +CME_ERROR: <cme_err> Purpose: Unsubscribe from a specific MQTT session topic. **Parameters:** <socket_ID> (MQTT client socket ID) • Unique integer value assigned by +KMQTTCFG, which identifies the MQTT client to the broker <topic_name> (Name of MQTT session topic to subscribe to) • ASCII string <status> (MQTT connection status) • 0—Connection aborted error • 1—Connection successful (CONNACK received from the MQTT broker) • 2—Subscribed to a topic successful (SUBACK received from the MQTT broker) • 3—Unsubscribed to a topic successful (UNSUBACK received from the MQTT broker) • 4—Message published successful (PUBACK received from the MQTT broker) • 5—Generic error • 6—Socket open successful <cme_err> (CME error code) • 910—CME_ER_BAD_SESSION_ID • 916—CME_ER_PARAMETER_INVALID_RANGE

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KPRIVKDELETE	**Delete private key from the index** **Usage:** • Execution: **AT+KPRIVKDELETE=<index>** Response: OK or +CME ERROR: <err> Purpose: Delete private key. • Query: **AT+KPRIVKDELETE=?** Response: +KPRIVKDELETE: (list of possible <index>es) OK Purpose: Display valid execution format and parameter values. **Parameters:** <index> (Stored private key index) • Valid range: 0–2

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KPRIVKSTORE	**Store private key associated to a local certificate** **Notes:** • Before using this command, it is highly recommended to configure the module for hardware flow control using AT+IFC. • The data session can end in the following ways: • Automatically, when <NbData> data bytes are sent or received, and the module returns to command state and returns OK. • Automatically, 10 seconds after "CONNECT" is received (i.e., the host most finish sending. • Additional methods to be supported in a future release: • By the host sending the end of file pattern "--EOF--Pattern--". • By the host sending "+++". • ATO is not available for this command. **Usage:** • Execution: **AT+KPRIVKSTORE=<index>[,<NbData>]** Response: CONNECT <File data> ← *Host sends <File_data> to store. See Examples (page 165).* <EOF_Pattern> ← *Host ends data session.* OK *or* +CME ERROR: <err> Purpose: Store private key. Note—An error will be returned if the <File_data> is not correct (e.g., if the key begin or key end patterns are missing or incorrect). • Query: **AT+KPRIVKSTORE?** Response: CONNECT private key,<index>,<NbData> ← *Private key with index=1* <File_data> [...] ← *Private keys with index=1–2* OK *or* +CME ERROR: <err> Purpose: Display details of the stored private keys. • Query List: **AT+KPRIVKSTORE=?** Purpose: Display valid execution format and parameter values. **Parameters:** <index> (Stored private key index) • Valid range: 0—2 <NbData> (Mandatory for both reading and writing) • Number of bytes to read or write • Valid range: 1—3000 <File data> (Data bytes) • String (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
KPRIVKSTORE (continued)	**Store private key associated to a local certificate (continued)** <EOF_pattern> (End of file pattern) • Note—This will be supported in a future release. (It does not appear as of the release date of this document.) • String • Valid value: "--EOF--Pattern--" **Example(s):** **AT+KPRIVKSTORE=0** CONNECT -----BEGIN RSA PRIVATE KEY----- 1234567890abcdefghijklmnopqrstuvwxyz -----END RSA PRIVATE KEY----- --EOF--Pattern-- OK **AT+KPRIVKSTORE=1** CONNECT -----BEGIN RSA PRIVATE KEY----- 1234567890!@#$%^&*() -----END RSA PRIVATE KEY----- --EOF--Pattern-- OK **AT+KPRIVKSTORE=2** CONNECT -----BEGIN RSA PRIVATE KEY----- 1234567890abcdefghijklmnopqrstuvwxyz -----END RSA PRIVATE KEY----- --EOF--Pattern-- OK **AT+KPRIVKSTORE?** CONNECT private key,0,36 -----BEGIN RSA PRIVATE KEY----- 1234567890abcdefghijklmnopqrstuvwxyz -----END RSA PRIVATE KEY----- private key,1,20 -----BEGIN RSA PRIVATE KEY----- 1234567890!@#$%^&*() -----END RSA PRIVATE KEY----- private key,2,36 -----BEGIN RSA PRIVATE KEY----- 1234567890abcdefghijklmnopqrstuvwxyz -----END RSA PRIVATE KEY----- OK

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KSSLCFG	**SSL configuration** Configure the SSL protocol for use with the module. **Usage:** • Execution: **AT+KSSLCFG=<option_id>,<option>** Response: If <option_id> = 0: AT+KSSLCFG=<option_id>,<TLS Version> OK or If <option_id> = 1: AT+KSSLCFG=<option_id>,<Random Seed> OK or If <option_id> = 2: AT+KSSLCFG=<option_id>,<Session Mode> OK Purpose: Configure the SSL protocol. • Query: **AT+KSSLCFG?** Response: +KSSLCFG:0,<TLS Version> +KSSLCFG:2,<Session Mode> OK Purpose: Display the SSL protocol configuration. • Query List: **AT+KSSLCFG=?** Purpose: Display valid execution format and parameter values. **Parameters:** <option_id> • Integer • Valid values: • 0—Specify a TLS version to be used for the hand shake • 1—Set up random seed • 2—Specify session mode <TLS Version> • Integer • Valid values: • 0—Highest possible • 3—TLS 1.2 <Random Seed> (String to be added into the entropy of the random number generator) • String <Session Mode> • Integer • Valid values: • 0—Automatic • 1—Always start a new session (not supported)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KSSLCRYPTO	**Cipher suite configuration** Configure the cipher suite used for an SSL profile. **Notes:** • Refer to Table 10-4 for the list of cipher suites supported by the RC71xx module. **Usage:** • Execution: **AT+KSSLCRYPTO=<profile_idx>,<mkey_algo>,<auth_algo>,<enc_algo>,<mac_algo>,<tls_ver>,<auth>[,<root_cert_idx>]** Response: OK Purpose: Configure the cipher suite used for an SSL profile. • Query: **AT+KSSLCRYPTO?** Response: +KSSLCRYPTO: <profile_idx>,<mkey_algo>,<auth_algo>,<enc_algo>,<mac_algo>,<tls_ver>,<auth>,<root_cert_idx>[…]] OK Purpose: Display the cipher suite used for an SSL profile. • Query List: **AT+KSSLCRYPTO=?** Purpose: Display valid execution format and parameter values. **Notification format:** **+KTCP_DATA: <session_id>,<ndata_available>[,<data>]** **Parameters:** <profile_idx> (Index of a set of parameters for configuring one SSL profile) • Integer <mkey_algo> (Key exchange algorithm selection) • Integer • 8—ECDHE <auth_algo> (Authentication algorithm selection) • Integer • 1—RSA • 2—ECDSA <enc_algo> (Encryption algorithm selection) • Integer • 16—AES-128-CCM • 32—AES-256-CCM • 256—AES-128-CCM-8 • 512—AES-256-CCM-8 • 8192—AES-128-GCM • 16384—AES-256-GCM <mac_algo> (Message authentication code for the algorithm selection) • Integer • 0—NULL • 4—SHA256 • 8—SHA384 <tls_ver> (Cipher suite version selection) • Integer • 4—TLS 1.2 (Continue on the next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
KSSLCRYPTO (continued)	**Cipher suite configuration (continued)** <auth> (Authentication) • String • 1—Authenticate server • 3—Mutual authentication <root_cert_idx> (Root certificate index • Integer • Valid range: • 0—3: Stored root certificate index. Defaults to 0 if not specified.

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KTCPCFG	**Configure TCP connection** Set or display the TCP connection configuration. **Notes:** • If the socket is defined as a <CLIENT> socket, <tcp_port> and <tcp_remote_address> define the port and IP address of the remote server to be connected. • For child session, the <data_mode> will be kept the same as the server socket's setting. • Connection timeout for TCP socket is ~9 seconds with 3 retransmissions with 3 seconds delay. • For the <restore_on_boot> parameter, only the first server session is restored. **Password required:** No **Usage:** • Execution: **AT+KTCPCFG=[<cnx_cnf>], <mode>, [<tcp_remote_address>], <tcp_port>[,[<source_port>][,[<data_mode>][,[<URC_ENDTCP_enable>][,[<af>][,[<cipher_suite>][,[<restore_on_boot>]]]]]]]** Response: +KTCPCFG: <session_id> OK *or* +CME ERROR: <err> Purpose: Configure a connection and receive a TCP session ID. • Query: **AT+KTCPCFG?** Response: +KTCPCFG: <session_id>,<status>,<cnx_cnf>,<mode>[,<serverID>], <tcp_remote_address>, <tcp_port>[,<source_port>], <data_mode>, <URC_ENDTCP_enable>, <af>, <cipher_index>[,<restore_on_boot>] [...] OK Purpose: Display the configurations for all TCP sessions. • Query List: **AT+KTCPCFG=?** Purpose: Display valid execution format and parameter values. **Parameters:** <cnx_cnf> (PDP context configuration) • Integer • Valid value: 1 <session_id> (TCP session index) • Integer • Maximum value: 12 <mode> (Connection mode) • 0—Client • 1—Server • 2—Child (generated by server sockets) • 3—Secure client <tcp_remote_address> (IP address string or explicit name of remote server) • String • For server configuration, this parameter is left blank <tcp_port> (TCP port number) • Listening port for a server configuration. • Valid range: 1–65535 (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KTCPCFG (continued)	**Configure TCP connection (continued)** <status> (Connection state of the selected socket) • 0—Disconnected • 1—Connected <serverID> (Server session ID index) • Only for sockets in Child mode • Integer value <source_port> (Local TCP port number) • Valid range: 0–65535 • For server configuration, this parameter is left blank <data_mode> (Enable/disable display of <data> in URC) • 0—Do not display (Default) • 1—Display (This option is not supported.) <URC_ENDTCP_enable> • Reserved for future use <af> (IP address family type used for connection) • Valid values: • 0—IPv4 • 1—IPv6 <cipher_index> (Cipher suite profile index to use for a secured socket) • Integer value • Defined by +KSSLCRYPTO <restore_on_boot> (Restore server session on boot (only for server socket)) • 0—First server session is not restored on boot. (Default) • 1—First server session is restored on boot.

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KTCPCLOSE	**Close current TCP connection** Close a TCP socket and, if no other sessions are running, then release the PDP context. **Notes:** • AT+KTCPDEL=<session_id> can be used to delete the socket configuration after it has been closed. **Password required:** No **Usage:** • Execution: **AT+KTCPCLOSE=<session_id>[,<closing_type>]** Response: OK *or* +CME ERROR: <err> NO CARRIER +KTCP_NOTIF: <session_id>,<tcp_notif> Purpose: Close the specified TCP connection. • Query List: **AT+KTCPCLOSE=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (TCP session index) • Integer • Maximum value: 12 <closing_type> (Method used to close the TCP connection) • 1—Close TCP connection properly. Data sent to the module by +KTCPSND will be send to the TCP server and acknowledged before the socket is closed. <tcp_notif> (Cause of TCP connection failure) • See +KTCPCNX for details

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KTCPCNX	**Start TCP connection** Connect to a remote server or listen to a bound port, depending on the selected mode of <session_id>. **Password required:** No **Usage:** • Execution: **AT+KTCPCNX=<session_id>** Response: OK *or* +CME ERROR: <err> +KTCP_NOTIF: <session_id>,<tcp_notif> Purpose: Start a connection on the specified TCP session. • Query List: **AT+KTCPCNX=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (TCP session index) • Integer • Maximum value: 12 <tcp_notif> (Cause of TCP connection failure) • 0—Network error • 1—No more sockets available; max. number already reached • 2—Memory problem • 3—DNS error • 4—TCP disconnection by the server or remote client • 5—TCP connection error • 6—Generic error • 7—Fail to accept client requests • 8—Data sending is OK but +KTCPSND was waiting more or less characters • 9—Bad session ID • 10—Session is already running • 11—All sessions are used • 12—Socket connection timeout error • 13—SSL connection error • 14—SSL initialization error

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KTCP_DATA (notification)	**Incoming Data through TCP connection—Unsolicited notification** Unsolicited notification indicating data is incoming. **Notes:** • As soon as the connection is established, the module can receive data through the TCP socket. This notification is sent when data are available in the receive buffer. • This notification is sent for each TCP packet received sequentially. Notification of the following received packet is sent only when the current notification has been read with a +KTCP_RCV command. • When the +KTCPCFG <data_mode> parameter is set to 1, <ndata_available> will range from 1 to 1500 in the URC. If the user application sends over 1500 bytes of data to the module, the module will display those data with several URCs. **Notification format:** **+KTCP_DATA: <session_id>,<ndata_available>[,<data>]** **Parameters:** <session_id> (TCP session index) • Integer • Maximum value: 12 <ndata_available> (Amount of data to be read) • Integer • For <data_mode>=0, maximum number of bytes to be read in the TCP receive buffer. • For <data_mode>=1, maximum number of bytes to be read in <data> <data> (Data, in octet) • String • Length of data is specified in <ndata_available>.

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KTCPDEL	**Delete configured TCP session** **Notes:** • Session must be closed using +KTCPCLOSE before using this command. **Password required:** No **Usage:** • Execution: **AT+KTCPDEL=<session_id>** Response: OK or +CME ERROR: <err> Purpose: Delete the specified TCP connection. • Query List: **AT+KTCPDEL=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (TCP session index) • Integer • Maximum value: 12

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KTCP_IND (notification)	**TCP status—Unsolicited notification** Notification of TCP session status. **Notification format:** **+KTCP_IND: <session_id>,<status>** **Parameters:** <session_id> (TCP session index) • Integer • Maximum value: 12 <status> (TCP session status) • 1—Session is set up and ready for operation

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KTCPRCV	**Receive data through TCP connection** Receive data through a previously opened TCP socket. **Notes:** • Before using this command, it is highly recommended to configure the module for hardware flow control using AT+IFC. • <ndata> indicates the max data number that the terminal wishes to receive. If the TCP socket contains more data than <ndata> bytes, then only <ndata> bytes will be received. If the TCP socket contains less data than <ndata> bytes, then only TCP socket's data will be received. • (To be supported in a future release.) <EOF_pattern> will be added at the end of data automatically. (i.e., The pattern is not included in the <ndata> bytes that are received. • When <ndata> (max value) bytes or only available data in the TCP socket have been received, the module returns to command state and returns OK. **Password required:** No **Usage:** • Execution: **AT+KTCPRCV=<session_id>,<ndata>** Response: CONNECT ...<EOF_pattern> ← *End of file string (to be supported in a future release)* OK or +KTCP_NOTIF: <session_id>,<tcp_notif> Purpose: Receive <ndata> bytes of data from the specified socket. • Query List: **AT+KTCPRCV=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (TCP session index) • Integer • Maximum value: 12 <ndata> (Number of bytes that device wants to receive) • Max value: 1400 <EOF_pattern> (End of file pattern) • Note—This will be supported in a future release. (It does not appear as of the release date of this document.) • String • Valid value: "--EOF--Pattern--" <tcp_notif> (Cause of TCP connection failure) • See +KTCPCNX for details

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KTCPSND	**Send data through TCP connection** Send data through a previously opened TCP socket. **Notes:** • Before using this command, it is highly recommended to configure the module for hardware flow control using AT+IFC. • All the data will be sent out ignoring <ndata>. If data sent is not equal to <ndata> then +KTCP_NOTIF will be displayed. • The data session can end in the following ways: • By the host sending the end of file pattern "--EOF--Pattern--". • By the host sending "+++". **Password required:** No **Usage:** • Execution: **AT+KTCPSND=<session_id>,<ndata>** Response: CONNECT OK or NO CARRIER +CME ERROR: <err> +KTCP_NOTIF: <session_id>,<tcp_notif> Purpose: Send <ndata> bytes of data to the specified socket. • Query List: **AT+KTCPSND=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (TCP session index) • Integer • Maximum value: 12 <ndata> (Number of bytes that device will send) • Max value: 1400 <tcp_notif> (Cause of TCP connection failure) • See +KTCPCNX for details

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KTCP_SRVREQ (notification)	**Incoming client connection request—Unsolicited notification** Notification received when a client requests a connection to the server. **Notes:** • This notification is sent when a client requests a connection to the server. The connection is automatically accepted. • The created session is driven as any other TCP session with its own session ID. Use +KTCPSND, +KTCPRCV, +KTCPCLOSE, etc. to provide the service associated to this TCP server. • The TCP server corresponding to the session ID is still able to receive connection requests from other clients. These requests are notified with +KTCP_SRVREQ. • The client IP address and port can also be checked using AT+KTCPCFG? after the client is connected to the TCP server. **Notification format:** **+KTCP_SRVREQ: <session_id>, <subsession_id>, <client_ip>, <client_port>** **Parameters:** <session_id> (TCP session index) • Integer • Maximum value: 12 <subsession_id> (Newly created TCP session index) • Integer • Maximum value: 12 <client_ip> (IP address of incoming socket) • String <client_port> (Port of the incoming client) • Valid range: 0–65535 **Example(s):** • Configure the module to TCP servers **AT+KTCPCFG=0,1,,179** +KTCPCFG: 1 OK **AT+KTCPCFG=0,1,,180** +KTCPCFG: 2 OK • Start the TCP servers - listen on port 179 **AT+KTCPCNX=1** OK - listen on port 180 **AT+KTCPCNX=2** OK (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KTCP_SRVREQ (notification) continued	**Incoming client connection request—Unsolicited notification (continued)** • Show the TCP servers' IP address—Incoming connection request from remote client, shows ip address and port of remote client **AT+CGPADDR** +CGPADDR: 0,"192.168.1.49" OK // incoming a connection request from "192.168.0.32" via listening port 179, the remote port is 4614 +KTCP_SRVREQ: 1,3,"192.168.0.32",4614 // incoming a connection request from "10.10.10.110" via listening port 180, the remote port is 4665 +KTCP_SRVREQ: 2,4,"10.10.10.110",4665 // incoming a connection request from the same ip via the same listening port, the remote port is 4668 +KTCP_SRVREQ: 2,5,"10.10.10.110",4668 // incoming a connection request from "192.168.1.117" via listening port 179, the remote port is 1739 +KTCP_SRVREQ: 1,6,"192.168.1.117",1739 // the connection of sub session id 4 (on listening port 180) is closed. +KTCP_NOTIF: 4,4 // incoming a connection request from "10.10.10.8" via listening port 180, the remote port is 4672 +KTCP_SRVREQ: 2,4,"10.10.10.8",4672

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KTCPSTART	**Start TCP connection in Direct Data Flow** Start a TCP connection in Direct Data Flow. **Notes:** • Command is used to send and receive data bytes through a TCP socket. • Before using this command, it is highly recommended to configure the module for hardware flow control using AT+IFC. • Only one +KTCPSTART session can be used. • Can be used in 07.10 multiplexer. • If the session is successfully connected by +KTCPCNX, this command does not restart the connection and the module directly enters direct data flow. • The data session can end in the following ways: • By the host sending the end of file pattern "--EOF--Pattern--". • By the host sending "+++". **Password required:** No **Usage:** • Execution: **AT+KTCPSTART=<session_id>** Response: CONNECT OK or +CME ERROR: an error occurs, syntax error +KTCP_NOTIF: <session_id>,<tcp_notif> : an error occurs Purpose: Start a TCP session in direct data flow. • Query: **AT+KTCPSTART?** Response: OK Purpose: • Query List: **AT+KTCPSTART=?** Response: OK Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (TCP session index) • Integer • Maximum value: 12 <tcp_notif> (Cause of TCP connection failure) • See +KTCPCNX for details

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KTCPSTAT	**Get TCP socket status** **Notes:** • Command returns +CME ERROR: 910 (Bad Session ID) for undefined <session_id>s. **Password required:** No **Usage:** • Execution (for all TCP session IDs): **AT+KTCPSTAT** Response: +KTCPSTAT: <session_id>, <status>, <tcp_notif>, <rem_data>, <rcv_data> [...] OK Purpose: Display socket statuses for all TCP sessions. • Execution (single TCP session ID): **AT+KTCPSTAT=<session_id>** Response: +KTCPSTAT: <status>, <tcp_notif>, <rem_data>, <rcv_data> OK Purpose: Display socket status for the specified TCP session. • Query: **AT+KTCPSTAT?** Response: OK Purpose: This command serves no purpose. It is included for compatibility purposes only. • Query List: **AT+KTCPSTART=?** Response: OK Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (TCP session index) • Integer • Maximum value: 12 <status> (TCP socket state) • 0—Socket not defined, use +KTCPCFG to create a TCP socket • 1—Socket is defined, but not used • 2—Socket is opening and connecting to the server, cannot be used • 3—Connection is up, socket can be used to send/receive data • 4—Connection is closing, cannot be used, wait for <status>=5 • 5—Socket is closed <tcp_notif> (TCP connection status) • -1—Socket/connection is OK • ≥0—TCP connection failure. See +KTCPCNX for details. <rem_data> (Remaining bytes in socket buffer, waiting to be sent) • String <rcv_data> (Received bytes, can be read with +KTCPRCV) • String

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KUDPCFG	**Configure UDP connection** Set or display UDP connection configuration. **Notes:** • For UDP socket in server mode, it is bound to a defined port number, incoming connection are notified by +KUDP_DATA. If remote address and port are given, they are saved for use in +KUDPSND. • Maximum <session_id> is 12. **Password required:** No **Usage:** • Execution: **AT+KUDPCFG=[<cnx_cnf>], <mode> [,[<port>] [,[<data_mode>][, [<udp_remote_address>] [, [<udp_port>] [, [<af>] [,[<cipher_suite>][, [<restore_on_boot>]]]]]]]]** Response: +KUDPCFG: <session_id> OK or +CME ERROR: <err> +KUDP_NOTIF: <session_id>, <udp_notif> Purpose: Configure a connection and receive a UDP session ID. • Query: **AT+KUDPCFG?** Response: +KUDPCFG: <session_id>, <cnx_cnf>, <mode>, <port>, <data_mode>, <udp_remote_address>, <udp_port>, <af>, <cipher_suite>, <restore_on_boot> [...] OK Purpose: Display the configurations for all UDP sessions. • Query List: **AT+KUDPCFG=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (UDP session index) • Integer • Maximum value: 12 <mode> (Connection mode) • 0—Client • 1—Server • 3—Secure client <port> • Valid range: 0–65535 (0 = random) <cnx_cnf> (PDP context configuration) • Integer • Valid value: 1 (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KUDPCFG (continued)	**Configure UDP connection (continued)** <udp_notif> (Cause of UDP connection failure) • 0—Network error • 1—No more sockets available; max number already reached • 2—Memory problem • 3—DNS error • 5—UDP connection error (Host unreachable) • 6—Generic error • 8—Data sending is OK but +KUDPSND was waiting more or less characters • 9—Bad session ID • 10—Session is already running • 11—All sessions are used • 13—UDP SSL connection error • 15—Bad server mode <data_mode> (Enable/disable display of <data> in URC) • 0—Do not display (Default) • 1—Display <data> in URC <udp_remote_address> (IP address string or explicit name of remote server) • String • Default is empty (given by +KUDPSND) <udp_port> (UDP peer port) • Valid range: 0–65535 • Given by +KUDPSND <af> (IP address family type used for connection) • Valid values: • 0—IPv4 • 1—IPv6 <cipher_suite> (Cipher suite profile index to use for a secured socket) • Defined by +KSSLCRYPTO <restore_on_boot> (Restore server session on boot (only for server socket)) • 0—First server session is not restored on boot. (Default) • 1—First server session is restored on boot.

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KUDPCLOSE	**Close current UDP connection** Close a UDP session and then, if no other sessions are running, release the PDP context. **Notes:** • This function will delete the session configuration if <keep_cfg> = 0 **Password required:** No **Usage:** • Execution: **AT+KUDPCLOSE=<session_id>[,<keep_cfg>]** Response: OK or +KUDP_NOTIF: <session_id>,<udp_notif> Purpose: Close the specified UDP connection. • Query List: **AT+KUDPCLOSE=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (UDP session index) • Integer • Maximum value: 12 <udp_notif> (Cause of UDP connection failure) • See +KUDPCFG for details <keep_cfg> (Delete/keep session configuration after closing) • 0—Delete the session configuration • 1—Keep the session configuration

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KUDP_DATA (notification)	**Incoming Data through UDP connection—Unsolicited notification** Unsolicited notification indicating data is incoming. **Notes:** • As soon as the UDP socket is created, the module can receive data through this socket. This notification is sent when data are available in the receive buffer. • This notification will be sent one time. When <data_mode> was set to 0 (do not display data in URC), the controlling software must read the buffer with +KUDPRCV to activate the notification again. • When <data_mode> was set to 1, <ndata_available> will range from 1 - 1500 in the URC. If the user application sends over 1500 bytes of data to the module, the module will display those data with several URCs. It is possible for other applications (e.g., from Windows) to send more than 1472 bytes UDP packets to the module but the packet will be segmented and reassembled by the network stack. • When <data_mode> is set to 1, URC +KUDP_RCV will not be displayed after +KUDP_DATA. • When <data_mode> was set to 1, the fields <udp remote address> and <udp remote port> will be displayed in URC +KUDP_DATA. When <data_mode> was set to 0, they will be displayed in URC +KUDP_RCV. **Notification format:** **+KUDP_DATA: <session_id>,<ndata_available>[,<udp_remote_address>, <udp_remote_port>, <data>]** **Parameters:** <session_id> (UDP session index) • Integer • Maximum value: 12 <ndata_available> (Amount of data to be read) • String <udp_remote_address> (IP address string of remote host) • String <udp_remote_port> (Remote UDP port) • Valid range: 0–65535 <data> (Data, in octet) • String • Length of data is specified in <ndata_available>.

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KUDPDEL	**Delete configured UDP session** **Notes:** • Session must be closed using +KUDPCLOSE before using this command. **Password required:** No **Usage:** • Execution: **AT+KUDPDEL=<session_id>** Response: OK or +CME ERROR: <err> Purpose: Delete the specified UDP connection. • Query List: **AT+KUDPDEL=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (UDP session index) • Integer • Maximum value: 12

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KUDP_IND (notification)	**UDP status—Unsolicited notification** Notification of UDP session status. **Notification format:** **+KUDP_IND: <session_id>,<status>** **Parameters:** <session_id> (UDP session index) • Integer • Maximum value: 12 <status> (UDP session status) • 1—Session is set up and ready for operation

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KUDPRCV	**Receive data through UDP connection** Receive data through a previously opened UDP socket. **Notes:** • Before using this command, it is highly recommended to configure the module for hardware flow control using AT+IFC. • <ndata> indicates the max data number that the terminal wishes to receive. If the UDP socket contains more data than <ndata> bytes, then only <ndata> bytes will be received and more data can be read by running this command again. • (To be supported in a future release.) <EOF_pattern> will be added at the end of data automatically. (i.e., The pattern is not included in the <ndata> bytes that are received.) • When <ndata> (max value) bytes or only available data in the UDP socket have been received, the module returns to command mode. **Password required:** No **Usage:** • Execution: **AT+KUDPRCV=<session_id>,<ndata>** Response: CONNECT ...<EOF_pattern> ← *End of file string (to be supported in a future release)* OK +KUDP_RCV: <udp_remote_address>, <udp_remote_port> or NO CARRIER +CME ERROR: <err> +KUDP_NOTIF: <session_id>,<udp_notif> Purpose: Receive <ndata> bytes of data from the specified socket. • Query List: **AT+KUDPRCV=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (UDP session index) • Integer • Maximum value: 12 <ndata> (Number of bytes that device wants to receive) • Max value: 1400 <EOF_pattern> (End of file pattern) • Note—This will be supported in a future release. (It does not appear as of the release date of this document.) • String • Valid value: "--EOF--Pattern--" <udp_remote_address> (IP address string of the remote host) • String <udp_remote_port> (Remote UDP port) • Valid range: 0–65535 <udp_notif> (Cause of UDP connection failure) • Integer • See +KUDPCFG for details

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KUDPSND	**Send data through UDP connection** Send data through a previously opened UDP connection. **Notes:** • Before using this command, it is highly recommended to configure the module for hardware flow control using AT+IFC. • All data will be sent out ignoring <ndata>. If data sent is not equal to <ndata> then +KUDP_NOTIF will be displayed. • The maximum transmission unit (MTU) is 1500 Bytes. • The <udp remote address> and <udp_port> are saved internally such that they can be omitted in subsequent calls of +KUDPSND. • The packet segmentation is controlled by +KIPOPT with <option_id>=0, and the maximum UDP packet size is limited by <send size v4> (1472 bytes) or <send size v6> (1452 bytes). Default value for both parameters is 1020 bytes. • The data session can end in the following ways: • By the host sending the end of file pattern "--EOF--Pattern--". • By the host sending "+++". **Password required:** No **Usage:** • Execution: **AT+KUDPSND=<session_id>, <udp_remote_address>, <udp_port>, <ndata>** Response: CONNECT OK or NO CARRIER +CME ERROR: <err> +KTCP_NOTIF: <session_id>,<udp_notif> Purpose: Send <ndata> bytes of data over the specified connection. • Query List: **AT+KUDPSND=?** Purpose: Display valid execution format and parameter values. **Parameters:** <session_id> (UDP session index) • Integer • Maximum value: 12 <udp_remote_address> (IP address string or explicit name of remote server) • String <udp_port> (UDP peer port) • Valid range: 1–65535 <ndata> (Number of bytes that device will send) • Max value: 1400 • Note—<ndata> does not include the end of file string (e.g., "--EOF--Pattern--", "+++") <udp_notif> (Cause of UDP connection failure) • See +KUDPCFG for details

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KURCCFG	**Enable/Disable Protocol Notifications (URCs)** Enable or disable specific protocol notifications (URCs). **Notes:** • Enabling/disabling +KTCP_NOTIF unsolicited notifications is only useful when in polling mode with +KTCPSTAT. • If notifications and/or indications are disabled, the URCs are discarded and not stored. • Can be used in 07.10 multiplexer. **Password required:** No **Usage:** • Execution: **AT+KURCCFG=<protoopt>,<noti_act>[,<indi_act>]** Response: OK Purpose: Enable/disable specified URC notifications and/or indications for the specified protocol. • Query: **AT+KURCCFG?** Response: +KURCCFG: (list of supported <protoopt>,<noti_act>,<indi_act>) OK Purpose: Display the status of URC notifications/indications for each protocol. • Query List: **AT+KURCCFG=?** Purpose: Display valid execution format and parameter values. **Parameters:** <protoopt> (Protocol option to enable/disable URC) • String format • "TCPC"—TCP client session • "TCPS"—TCP server session • "UDPC"—UDP client session • "UDPS"—UDP server session • "HTTP"—HTTP client session • "HTTPS"—HTTPS client session • "TCP"—Both TCP client and server sessions • "UDP"—Both UDP client and server sessions <noti_act> (Enable/disable URC notifications) • 0—Disable URC • 1—Enable URC (such as +KTCP_NOTIF) <indi_act> (Enable/disable URC indications) • 0—Disable URC • 1—Enable URC (such as +KTCP_SRVREQ, +KTCP_IND, +KTCP_DATA, +KUDP_DATA, +KUDP_RCV, etc.) (Continued on next page)

Table 10-3: Protocol Command Details (Continued)

Command	Description
+KURCCFG (continued)	**Enable/Disable Protocol Notifications (URCs) (continued)** **Example(s):** • Disable URC notifications: AT+KURCCFG="TCP",0 OK • Test and read commands: AT+KURCCFG=? +KURCCFG: ("TCPC","TCPS","UDPC","UDPS", "HTTP","HTTPS","TCP","UDP"),(0,-1),(0-1) OK AT+KURCCFG? +KURCCFG: "TCPC",1,1 +KURCCFG: "TCPS",1,1 +KURCCFG: "UDPC",1,1 +KURCCFG: "UDPS",1,1 +KURCCFG: "HTTP",1,1 +KURCCFG: "HTTPS",1,1 OK

Table 10-4: Supported Cipher Suites

NIST Name	<mkey_algo>	<auth_algo>	<enc_algo>	<mac_algo>
TLS-ECDHE-RSA-WITH-AES-128-GCM-SHA256	ECDHE	RSA	AES-128-GCM	SHA256
TLS-ECDHE-ECDSA-WITH-AES-128-GCM-SHA256	ECDHE	ECDSA	AES-128-GCM	SHA256
TLS-ECDHE-ECDSA-WITH-AES-256-GCM-SHA384	ECDHE	ECDSA	AES-256-GCM	SHA384
TLS-ECDHE-ECDSA-WITH-AES-128-CCM	ECDHE	ECDSA	AES-128-CCM	NULL
TLS-ECDHE-ECDSA-WITH-AES-256-CCM	ECDHE	ECDSA	AES-256-CCM	NULL
TLS-ECDHE-ECDSA-WITH-AES-128-CCM-8	ECDHE	ECDSA	AES-128-CCM-8	NULL
TLS-ECDHE-ECDSA-WITH-AES-256-CCM-8	ECDHE	ECDSA	AES-256-CCM-8	NULL

10.4 Sample IP Application Command Usage

10.4.1 How to Use TCP Commands

Table 10-5: Client Mode

Example	Description
AT+KTCPCFG=1,0,"www.google.com",8 0 +KTCPCFG: 1 OK	Set IP address and port number Returns session ID
AT+KTCPCNX=1 OK	Initiate the connection.
AT+KTCPSND=1,18 CONNECT …Data send… OK	Send data after CONNECT. To end the data transfer and exit data mode, send the EOF pattern ("+++").
+KTCP_DATA: 1,1380	+KTCP_DATA notification
AT+KTCPRCV=1, 1380 CONNECT … data… --EOF--Pattern-- OK	DATA read **Note—**The end of pattern string is to be supported in a future release.
+KTCP_DATA: 1,1380	+KTCP_DATA notification
AT+KTCPRCV=1,1380 CONNECT … a lot of data… --EOF--Pattern-- OK	DATA read **Note—**The end of pattern string is to be supported in a future release.
+KTCP_DATA: 1,1380	+KTCP_DATA notification
AT+KTCPCLOSE=1,1 OK	Close session 1
AT+KTCPDEL=1 OK	Delete session 1
AT+KTCPCFG? OK	No session is available

A TCP server is emulated in the following example. The server listens to port 13 and returns the date for each connection.

Table 10-6: Server Mode

Example	Description
AT+KTCPCFG=1,1,,13 +KTCPCFG: 1 OK	Set TCP listener and port number Returns session 1
AT+KTCPCNX=1 OK	Initiate the server.
AT+CGPADDR +CGPADDR: 0,"10.35.125.89" OK	Get the IP address to initiate a connection request with a client
+KTCP_SRVREQ: 1,2	A client requests a connection (sub-session 2)
AT+KTCPSND=2,15 CONNECT …Date and time… OK	Data is sent to the client read (based on sub-session 2)
+KTCP_SRVREQ: 1,3	Another client requests a connection (sub-session 3); child mode for session 3
+KTCP_NOTIF: 2,4	Client (sub-session 2) closes the connection
AT+KTCPSND=3,15 CONNECT …Date and time… OK	Data is sent to the client
+KTCP_DATA: 3,6	Data received from the client (sub-session 3)
AT+KTCPRCV=3,6 CONNECT … Data… --EOF--Pattern-- OK	Read data received from client **Note—**The end of pattern string is to be supported in a future release.
AT+KTCPCLOSE=3,1 OK	Close client sub-session 3 and then sub-session 3 is deleted automatically
AT+KTCPCLOSE=1,1 OK	Close server session 1
AT+KTCPDEL=1 OK	Delete session 1

Table 10-7: Polling for the Status of a Socket

Example	Description
AT+KTCPCFG=1,0,"www.google.com",8 0 +KTCPCFG: 1 OK	The set TCP Server address and port number, Returns the session ID
AT+KURCCFG="TCP",0 OK	Disable TCP unsolicited messages

Table 10-7: Polling for the Status of a Socket (Continued)

Example	Description
AT+KTCPCNX=1 OK	Initiate the connection, use session 1
AT+KTCPSTAT=1 +KTCPSTAT : 3,-1,0,0 OK	Poll the connection status Connection is UP
AT+KTCPSND=1,3000	Send data on socket 1, we expect to send 3000 bytes but you can send less.
CONNECT	You can send data after CONNECT
…Data send… OK	To end the data transfer and exit data mode, send the EOF pattern ("+++").
AT+KTCPSTAT=1	Poll the connection status
+KTCPSTAT : 3,-1,1234,0 OK	Connection is UP, there are 1234 bytes not yet sent
AT+KTCPSTAT=1 +KTCPSTAT : 3,-1,100,0 OK	Poll the connection status Connection is UP, there are 100 bytes not yet sent
AT+KTCPSTAT=1 +KTCPSTAT : 3,-1,0,0 OK	Poll the connection status Connection is UP, all bytes have been sent
AT+KTCPSTAT=1 +KTCPSTAT : 3,-1,0,320 OK	Poll the connection status Connection is UP, 320 bytes are available for reading
AT+KTCPRCV=1,320 CONNECT … a lot of data… --EOF--Pattern-- OK	Read 320 bytes on socket 1 Data are sent after CONNECT **Note—**The end of pattern string is to be supported in a future release.
AT+KTCPCLOSE=1,1 OK	Close session 1
AT+KTCPDEL=1 OK	Delete session 1

Table 10-8: End to End TCP Connection

Example	Description
AT+KTCPCFG=1,0,"www.google.com",8 0 +KTCPCFG: 1 OK	The set TCP Server address and port number Returns session ID
AT+KTCPSTART=1	Initiate the connection, use session 1
CONNECT	Message CONNECT: connection to the server is established, you can send data

Table 10-8: End to End TCP Connection (Continued)

Example	Description
...Data sent.....Data received.....Data sent... ...Data sent.....Data received.....Data sent... +++ OK	Use +++ to enter in command mode
AT+KTCPCLOSE=1,1 OK	Use KTCPCLOSE to close the session
AT+KTCPDEL=1 OK	Delete the configured session

Table 10-9: Error Case for End to End TCP Connection

Example	Description
AT+KTCPSTART=1 NO CARRIER +KTCP_NOTIF: 1,<tcp_notif>	Try to initiate the connection Connection fails, see the value of <tcp_notif>
AT+KTCPSTART=1 CONNECT ...Data sent.....Data received.....Data sent... ...Data sent.....Data received.....Data sent... NO CARRIER	Initiate the connection Exchange some data
+KTCP_NOTIF: 1,<tcp_notif>	An error occurs during connection (network lost, the server closed, etc.)

10.4.2 How to Use UDP Specific Commands

Table 10-10: Client Mode

Example	Description
AT+KUDPCFG=1,0,1025 +KUDPCFG: 1 OK +KCNX_IND: 1,1,0 +KUDP_IND: 1,1	Create a new UDP socket with port number (returned session 1) with the parameters associated to the connection profile id number 1
AT+KUDPSND=1,”213.41.22.60”,1025,1 0 CONNECT …Data Sent… --EOF--Pattern-- OK	Send UDP data after “CONNECT”
+KUDP_DATA: 1,10	Received notification that indicates the presence of 10 bytes in the socket
AT+KUDPRCV=1,5 CONNECT 12345--EOF—Pattern-- OK +KUDP_RCV: "213.41.22.60",1025 +KUDP_DATA: 1,5	Try to read 5 bytes from session 1 **Note—**The end of pattern string is to be supported in a future release. Received notification that indicates the presence of 5 bytes in the socket
AT+KUDPRCV=1,5 CONNECT 67890--EOF—Pattern-- OK +KUDP_RCV: “213.41.22.60”,1025	Try to read 5 bytes from session 1 **Note—**The end of pattern string is to be supported in a future release.
AT+KUDPCLOSE=1 OK	Close the UDP session 1
AT+KUDPDEL=1 OK	Delete session 1

10.4.3 How to Use UDP and TCP Simultaneously

Table 10-11: Client Mode

Example	Description
AT+KUDPCFG=1,0,1025 +KUDPCFG: 1 OK +KCNX_IND: 1,1,0 +KUDP_IND: 1,1	Create a new UDP socket with port number (returned session 1) with the parameters associated with the connection profile id number 1
AT+KUDPSND=1,”213.41.22.60”,1025,1 0 CONNECT …Data Sent… --EOF--Pattern-- OK	Send UDP data after “CONNECT” **Note—**The end of pattern string is to be supported in a future release.
AT+KTCPCFG=1,0,"www.google.com",8 0 +KTCPCFG: 2 OK	Set TCP Server address and port number Returns session ID
AT+KTCPCNX=2 OK	Initiate the connection
AT+KTCPSND=2,10 CONNECT …Data send… OK +KTCP_DATA: 2,10	Send data with "+++" string at the end
AT+KTCPRCV=2,10 CONNECT … data… --EOF--Pattern-- OK	DATA read **Note—**The end of pattern string is to be supported in a future release.
AT+KUDPCLOSE=1 OK	Close the UDP session 1
AT+KTCPCLOSE=2,1 OK	Close the TCP session 2
AT+KTCPDEL=2 OK	Delete the TCP session 2

11: Unsolicited Message Commands

11.1 Introduction

This chapter shows the commands related to USL.

11.2 Command summary

Table 11-1 lists the commands described in this chapter:

Table 11-1: Unsolicited Message Commands

Command	Description	Page
!MUSLEN	Enable or disable unsolicited message feature	200

11.3 Command reference

Table 11-2: Unsolicited Message Command Details

Command	Description
!MUSLEN	**Enable or disable unsolicited message feature** This command is used to enable or disable the unsolicited message feature. **Usage:** • Execution: **AT!MUSLEN=<enable>** Response: OK *or* ERROR Purpose: Enable the unsolicited message feature. • Query: **AT!MUSLEN?** Response: !MUSLEN: <enable> OK Purpose: Query the current feature's status. • Query List: **AT!MUSLEN=?** Purpose: Return the execution command format and the supported parameter values. **Parameters:** <enable> (Enable or disable the feature) • Valid values: • 0—Disable • 1—Enable

12: Supported GSM/WCDMA AT Commands

This chapter identifies standard AT commands that are supported by most Sierra Wireless devices. These commands:

- Control serial communications over an asynchronous interface (*ITU-T Serial Asynchronous Dialling and Control (Recommendation V.250*), available on the International Telecommunication Union web site, www.itu.int).
  See Table 12-1 below.
- Control SMS functions for devices on GSM/WCDMA networks (*3GPP TS 27.005*, available on the 3GPP web site, www.3gpp.org)
  See Table 12-2 on page 203.
- Control devices operating on GSM/WCDMA networks (*3GPP TS 27.007,* available on the 3GPP web site, www.3gpp.org)
  See Table 12-3 on page 204.

The tables below identify whether each command is supported on Sierra Wireless UMTS devices. An “N/A” in the Supported column of the table indicates that the command is related to a feature (such as voice) that is not available on the modems.

Commands that are partially supported include descriptions identifying any limitations on command usage. Also, some commands are described in more detail in other chapters—the descriptions for these commands link to those detailed entries (for example, **&V** in Table 12-1 on page 201).

Table 12-1: Supported ITU-T Recommendation V.250 AT Commands

Command	Description	Supported ✔=Yes; ✘=No
Commands		
&C	Set Data Carrier Detected (Received line signal detector) function mode	✔
&D	Set Data Terminal Ready function mode	✔
&F	Set all current parameters to manufacturer’s defaults	✔
&K	Flow control	✘ To set flow control, use AT+IFC instead.
&S	Set DSR signal	✘
&T	Auto tests	✘
&V	Return operating mode AT configuration parameters	✔
&W	Store current parameter to user-defined profile	✔
+++	Switch from Data Mode to Command Mode	✔ For details, see +++ on page 16
+DR	V42bis data compression report	✘
+DS	V42bis data compression	✘

Table 12-1: Supported ITU-T Recommendation V.250 AT Commands (Continued)

Command	Description	Supported ✔=Yes; ✘=No
+GCAP	Request complete TA capabilities list	✘
+GMI	Request manufacturer identification	✔
+GMM	Request TA model identification	✔
+GMR	Request TA revision identification	✔
+GOI	Request global object identification	✘
+GSN	Request TA serial number identification	✘
+ICF	Set TE-TA control character framing	✔
+IFC	Set TE-TA local data flow control	✔
+ILRR	Set TE-TA local rate reporting mode	✘
+IPR	*Note: Set fixed local rate (default rate is 115200).*	✔
A	Answer incoming call	✘
A/	Re-issues last AT command given	✘
D	Dial	✔
D><MEM><N>	Originate call to phone number in memory <MEM>	✘
D><N>	Originate call to phone number in current memory	✘
D><STR>	Originate call to phone number in memory which corresponds to alphanumeric field <STR>	✘
DL	Redial last telephone number used	✘
E	Set command echo mode	✔
H	Disconnect existing connections	✔
I	Display product identification information	✔ For details, see !HWID on page 37
L	Set monitor speaker loudness	✘
M	Set monitor speaker mode	✘
O	Switch from command mode to data mode	✔
P	Select pulse dialing	✘
Q	Set Result code presentation mode	✔
S0	Set number of rings before automatically answering the call	✘
S10	Set disconnect delay after indicating the absence of data carrier	✘
S3	Set command line termination character	✘

Table 12-1: Supported ITU-T Recommendation V.250 AT Commands (Continued)

Command	Description	Supported ✔=Yes; ✘=No
S4	Set response formatting character	✘
S5	Set command line editing character	✘
S6	Set pause before blind dialing	✘
S7	Set number of seconds to wait for connection completion	✘
S8	Set number of seconds to wait when comma dial modifier used	✘
S11	Query/set DTMF dialing speed	✘
T	Select tone dialing	✘
V	Set result code format mode	✘
X	Set connect result code format and call monitoring	✘
Z	Set all current parameters to user-defined profile	✘
Result Codes		
OK	Acknowledges execution of a command	✔
CONNECT	A connection has been established	✔
RING	Unsolicited notification of an incoming call signal from the network	✔
NO CARRIER	The connection has been terminated or the attempt to establish a connection failed	✔
ERROR	Command not recognized, command line maximum length exceeded, parameter value invalid, or other problem with processing the command line	✔
NO DIALTONE	No dial tone detected	✔
BUSY	Engaged (busy) signal detected	✔

Table 12-2: Supported 27.005 AT Commands

Command	Description	Supported ✔=Yes; ✘=No
+CBM	Cell broadcast message directly displayed	✔
+CBMI	Cell broadcast message stored in memory at specified <index> location	✔
+CDS	SMS status report after sending a SMS	✔
+CDSI	Incoming SMS status report	✔
+CMGC	Send command	✔
+CMGD	Delete message	✔
+CMGF	Message format	✔
+CMGL	List messages	✔

Table 12-2: Supported 27.005 AT Commands (Continued)

Command	Description	Supported ✔=Yes; ✘=No
+CMGR	Read message	✔
+CMGS	Send message	✔
+CMGW	Write message to memory	✔
+CMMS	More messages to send	✔
+CMS ERROR: <err>	SMS error (mobile or network error)	✔
+CMSS	Send message from storage	✔
+CMT	Incoming message directly displayed	✔
+CMTI	Incoming message stored in <mem> (“SM”—SIM message storage) at location <index>	✔
+CNMA	New message acknowledgment to mobile equipment	✔
+CNMI	New message indications to TE	✔
+CPMS	Preferred message storage	✔
+CRES	Restore settings	✘
+CSAS	Save settings	✘
+CSCA	Service center address	✔
+CSCB	Select cell broadcast message types	✔
+CSDH	Show text mode parameters	✔
+CSMP	Set text mode parameters	✔
+CSMS	Select message service	✔

Table 12-3: Supported 27.007 AT Commands

Command	Description	Supported ✔=Yes; ✘=No
C	ITU T V.24 circuit 109 carrier detect signal behavior command	✘
+CACM	Accumulated call meter	✘
+CACSP	Voice Group or Voice Broadcast Call State Attribute Presentation	✘
+CAEMLPP	eMLPP Priority Registration and Interrogation	✘
+CAHLD	Leave an ongoing Voice Group or Voice Broadcast Call	✘
+CAJOIN	Accept an incoming Voice Group or Voice Broadcast Call	✘
+CALA	Alarm	✘
+CALCC	List current Voice Group and Voice Broadcast Calls	✘
+CALD	Delete alarm	✘

Table 12-3: Supported 27.007 AT Commands (Continued)

Command	Description	Supported ✔=Yes; ✘=No
+CALM	Alert sound mode	✘
+CAMM	Accumulated call meter maximum	✘
+CANCHEV	NCH Support Indication	✘
+CAOC	Advice of Charge	✘
+CAPD	Postpone or dismiss an alarm	✘
+CAPTT	Talker Access for Voice Group Call	✘
+CAREJ	Reject an incoming Voice Group or Voice Broadcast Call	✘
+CAULEV	Voice Group Call Uplink Status Presentation	✘
+CBC	Battery charge	✘
+CBST	Select bearer service type	✘
+CCCM	Current call meter value	✘
+CCFC	Call forwarding number and conditions	✘
+CCIOTOPT	CIoT optimization configuration	✔
+CCLK	Clock	✔
+CCUG	Closed user group	✘
+CCWA	Call waiting	✘
+CCWE	Call Meter maximum event	✘
+CDIP	Called line identification presentation	✘
+CDIS	Display control	✘
+CEER	Extended error report	✔
+CEMODE	UE modes of operation for EPS	✔
+CEREG	EPS network registration status Note: Command implementation based on 3GPP 27.007 rel 8.11.0.	✔
+CFUN	Set phone functionality Format • +CFUN = [<fun> [, <rst>]] Limitations • Valid <fun> values: ◦ 0 (minimum functionality, low power draw) ◦ 1 (full functionality, high power draw) ◦ 4 (Airplane mode, low power draw)	✔
+CGACT	PDP context activate or deactivate	✔
+CGANS	Manual response to a network request for PDP context activation	✘

Table 12-3: Supported 27.007 AT Commands (Continued)

Command	Description	Supported ✔=Yes; ✘=No
+CGAPNRC	APN rate control	✔
+CGATT	PS attach or detach	✔
+CGAUTO	Automatic response to a network request for PDP context activation	✘
+CGCLASS	GPRS mobile station class	✘
+CGCLOSP	Configure local octet stream PAD parameters	✘
+CGCMOD	PDP Context Modify	✔
+CGCONTRDP	PDP Context Read Dynamic Parameters	✔
+CGDATA	Enter data state	✔
+CGDCONT	Define PDP Context	✔ For details, see +CGDCONT.
+CGDSCONT	Define Secondary PDP Context	✔
+CGEQMIN	3G Quality of Service Profile (Minimum acceptable)	✘
+CGEQNEG	3G Quality of Service Profile (Negotiated)	✘
+CGEQOS	Define EPS Quality of Service	✔
+CGEQOSRDP	EPS quality of service read dynamic parameters	✔
+CGEQREQ	3G Quality of Service Profile (Requested)	✘
+CGEREP	Packet Domain event reporting	✔
+CGEV	GPRS network event indication	✔
+CGMI	Request manufacturer identification	✔
+CGMM	Request model identification	✔
+CGMR	Request revision identification	✔
+CGPADDR	Show PDP address	✔
+CGQMIN	Quality of Service Profile (Minimum acceptable)	✘
+CGQREQ	Quality of Service Profile (Requested)	✘
+CGREG	GPRS network registration status	✘
+CGSCONTRDP	Secondary PDP Context Read Dynamic Parameters	✔
+CGSMS	Select service for MO SMS messages	✘
+CGSN	Request product serial number identification	✔
+CGTFT	Traffic Flow Template	✔
+CGTFTRDP	Traffic Flow Template Read Dynamic Parameters	✔

Table 12-3: Supported 27.007 AT Commands (Continued)

Command	Description	Supported ✔=Yes; ✘=No
+CHLD	Call related supplementary services	✔
+CHSA	HSCSD non-transparent asymmetry configuration	✘
+CHSC	HSCSD current call parameters	✘
+CHSD	HSCSD device parameters	✘
+CHSR	HSCSD parameters report	✘
+CHST	HSCSD transparent call configuration	✘
+CHSU	HSCSD automatic user initiated upgrading	✘
+CHUP	Hangup call	✘
+CIEV	Indicator event	✘
+CIMI	Request international mobile subscriber identity	✔
+CIND	Indicator control	✔
+CIPCA	Initial PDN context activation	✔
+CKEV	Key press or release event	✘
+CKPD	Keypad control	✘
+CLAC	List all available AT commands	✘
+CLAE	Language Event	✘
+CLAN	Set Language	✘
+CLCC	List current calls	✘
+CLCK	Facility lock	✔
+CLIP	Calling line identification presentation	✘
+CLIR	Calling line identification restriction	✘
+CLVL	Set/return internal loudspeaker volume	✘
+CMAR	Master Reset	✘
+CME ERROR: <err>	Mobile Termination error result code	✔
+CMEC	Mobile Termination control mode	✔
+CMEE	Report Mobile Termination error	✔ For details, see +CMEE on page 28
+CMER	Mobile Termination event reporting	✘
+CMOD	Call mode	✘
+CMUT	Enable/disable uplink voice muting	✘

Table 12-3: Supported 27.007 AT Commands (Continued)

Command	Description	Supported ✔=Yes; ✘=No
+CMUX	Multiplexing mode	✔ (When MUX mode is configured on UART interface.)
+CNUM	Subscriber number	✔
+COLP	Connected line identification presentation	✘
+COPN	Read operator names	✔
+COPS	Operator selection	✔
+CPAS	Phone activity status	✘
+CPBF	Find phonebook entries	✘
+CPBR	Read phonebook entries	✘
+CPBS	Select phonebook memory storage	✘
+CPBW	Write phonebook entry	✘
+CPIN	Enter PIN	✔
+CPLS	Preferred PLMN list selection	✔
+CPOL	Preferred operator list	✔
+CPROT	Enter protocol mode	✘
+CPUC	Price per unit and currency table	✘
+CPWC	Power class	✘
+CPWD	Change password	✔
+CR	Service reporting control	✘
+CRC	Cellular result codes	✘
+CREG	Network registration	✔
+CRING	Incoming call type	✔
+CRLP	Radio link protocol	✘
+CRMP	Ring Melody Playback	✘
+CRSL	Ringer sound level	✘
+CRSM	Restricted SIM access	✔
+CRTDCP	Reporting of terminating data through the control plane	✔
+CCIOTOPT	CIoT optimization configuration	✔
+CSCC	Secure control command	✘

Table 12-3: Supported 27.007 AT Commands (Continued)

Command	Description	Supported ✔=Yes; ✘=No
+CSCON	Signaling connection status	✔
+CSCS	Select TE character set	✔
+CSDF	Settings date format	✘
+CSGT	Set Greeting Text	✘
+CSIL	Silence Command	✘
+CSIM	Generic SIM access	✔
+CSNS	Single numbering scheme	✘
+CSODCP	Sending of originating data through the control plane	✔
+CSQ	Signal quality	✔
+CSSN	Supplementary service notifications	✘
+CSTA	Select type of address	✘
+CSTF	Settings time format	✘
+CSVM	Set Voice Mail Number	✘
+CTFR	Call deflection	✘
+CTZR	Time Zone Reporting	✔
+CTZU	Automatic Time Zone Update	✔
+CUSD	Unstructured supplementary service data	✘
+CV120	V.120 rate adaptation protocol	✘
+CVHU	Voice Hangup Control	✘
+CVIB	Vibrator mode	✘
D	ITU T V.25ter [14] dial command	✔
D*99#	Sets up a packet data call (PDP context) based on profile ID #1	✔
D*99*<n>#**	Sets up a packet data call (PDP context) based on profile ID #<n> (<n> is the <cid> in the +CGDCONT command)	✔
+VTD	Tone duration	✘
+VTS	DTMF and arbitrary tone generation	✘
+WS46	PCCA STD 101 [17] select wireless network	✘

13: Band Definitions

Some commands described in this document include input and/or output ‘band’ parameters, where the value is a 3GPP band number (Table 13-1 on page 210). Commands using this table:

- !ANTSEL on page 18

Note: Band support is product-specific—refer to the RC71xx Product Technical Specification for details.

Table 13-1: 3GPP Bands[a,b]

3GPP Band	Frequency ranges (MHz)		3GPP Band	Frequency ranges (MHz)	
	Tx	Rx		Tx	Rx
1	1920–1980	2110–2170	**30**	2305–2315	2350–2360
2	1850–1910	1930–1990	**31**	452.5–457.5	462.5–467.5
3	1710–1785	1805–1880	**32**	n/a	1452–1496
4	1710–1755	2110–2155	**33**	1900–1920	
5	824–849	869–894	**34**	2010–2025	
6	830–840	875–885	**35**	1850–1910	
7	2500–2570	2620–2690	**36**	1930–1990	
8	880–915	925–960	**37**	1910–1930	
9	1749.9–1784.9	1844.9–1879.9	**38**	2570–2620	
10	1710–1770	2110–2170	**39**	1880–1920	
11	1427.9–1447.9	1475.9–1495.9	**40**	2300–2400	
12	699–716	729–746	**41**	2496–2690	
13	777–787	746–756	**42**	3400–3600	
14	788–798	758–768	**43**	3600–3800	
15	Reserved	Reserved	**44**	703–803	
16	Reserved	Reserved	**45**	1447–1467	
17	704–716	734–746	**46**	5150–5925	
18	815–830	860–875	**47**	5855–5925	
19	830–845	875–890	**48**	3550–3700	
20	832–862	791–821	**49**	3550–3700	
21	1447.9–1462.9	1495.9–1510.9	**50**	1432–1517	
22	Reserved	Reserved	**51**	1427–1432	
23	2000–2020	2180–2200	**52**	3300–3400	

Table 13-1: 3GPP Bands[a,b] (Continued)

3GPP Band	Frequency ranges (MHz)		3GPP Band	Frequency ranges (MHz)	
	Tx	Rx		Tx	Rx
24	1626.5–1660.5	1525–1559	**53-64**	Reserved	Reserved
25	1850–1915	1930–1995	**65**	1920–2010	2110–2200
26	814–849	859–894	**66**	1710–1780	2110–2200
27	807–824	852–869	**67-70**	Reserved	Reserved
28	703–748	758–803	**71**	663–698	617–652
29	n/a	717–728			

a. For CDMA bands, use these equivalents: BC0 (Band 5), BC1 (Band 2), BC10 (Band 6).
b. Commands using this table are identified in the chapter introduction.

14: ASCII Table/CME Error Codes

Table 14-1: ASCII Values

Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex
NUL	0	00	SP	32	20	@	64	40	‘	96	60
SOH	1	01	!	33	21	A	65	41	a	97	61
STX	2	02	“	34	22	B	66	42	b	98	62
ETX	3	03	#	35	23	C	67	43	c	99	63
EOT	4	04	$	36	24	D	68	44	d	100	94
ENQ	5	05	%	37	25	E	69	45	e	101	95
ACK	6	06	&	38	26	F	70	46	f	102	96
BEL	7	07	’	39	27	G	71	47	g	103	97
BS	8	08	(	40	28	H	72	48	h	104	98
HT	9	09	)	41	29	I	73	49	i	105	99
LF	10	0A	*	42	2A	J	74	4A	j	106	6A
VT	11	0B	+	43	2B	K	75	4B	k	107	6B
FF	12	0C	,	44	2C	L	76	4C	l	108	6C
CR	13	0D	-	45	2D	M	77	4D	m	109	6D
SO	14	0E	.	46	2E	N	78	4E	n	110	6E
SI	15	0F	/	47	2F	O	79	4F	o	111	6F
DLE	16	10	0	48	30	P	80	50	p	112	70
XON	17	11	1	49	31	Q	81	51	q	113	71
DC2	18	12	2	50	32	R	82	52	r	114	72
XOFF	19	13	3	51	33	S	83	53	s	115	73
DC4	20	14	4	52	34	T	84	54	t	116	74
NAK	21	15	5	53	35	U	85	55	u	117	75
SYN	22	16	6	54	36	V	86	56	v	118	76
ETB	23	17	7	55	37	W	87	57	w	119	77
CAN	24	18	8	56	38	X	88	58	x	120	78
EM	25	19	9	57	39	Y	89	59	y	121	79
SUB	26	1A	:	58	3A	Z	90	5A	z	122	7A
ESC	27	1B	;	59	3B	[	91	5B	{	123	7B
FS	28	1C	<	60	3C	\	92	5C	|	124	7C
GS	29	1D	=	61	3D	]	93	5D	}	125	7D
RS	30	1E	>	62	3E	^	94	5E	~	126	7E
US	31	1F	?	63	3F	_	95	5F	DEL	127	7F

Table 14-2: CME Error Codes

<err> Code	Meaning
0	Phone failure
1	No connection to phone
2	Phone?adapter link reserved
3	Operation not allowed
4	Operation not supported
5	PH-SIM PIN required
6	PH-FSIM PIN required
7	PH-FSIM PUK required
10	SIM not inserted
11	SIM PIN required
12	SIM PUK required
13	SIM failure
14	SIM busy
15	SIM wrong
16	Incorrect password
17	SIM PIN2 required
18	SIM PUK2 required
20	Memory full
21	Invalid index
22	Not found
23	Memory failure
24	Text string too long
25	Invalid characters in text string
26	Dial string too long
27	Invalid characters in dial string
30	No network service
31	Network timeout
32	Network not allowed - emergency call only
40	Network personalization PIN required
41	Network personalization PUK required
42	Network subset personalization PIN required

Table 14-2: CME Error Codes (Continued)

<err> Code	Meaning
43	Network subset personalization PUK required
44	Service provider personalization PIN required
45	Service provider personalization PUK required
46	Corporate personalization PIN required
47	Corporate personalization PUK required
49	EAP method not supported
50	Incorrect parameters
51	Parameter length error for all Auth commands
52	Temporary error for all auth cmds
100	Unknown error
103	Illegal Mem_Store
106	Illegal ME
107	GPRS services not allowed
111	PLMN not allowed
112	Location area not allowed
113	Roaming not allowed in this location area
132	Service option not supported
133	Requested service option not subscribed
134	Service option temporarily out of order
148	Unspecified GPRS error
149	PDP authentication failure
150	Invalid mobile class
257	network rejected supserv request
258	retry operation
259	invalid deflected to number
260	deflected to own number
261	unknown subscriber
262	service not available
263	unknown class
264	unknown network message
273	Minimum TFT per PDP address error

Table 14-2: CME Error Codes (Continued)

<err> Code	Meaning
274	Duplicate TFT eval prec index
275	Invalid TFT param combination
320	Call Index Error
321	Call State Error
322	Sys State Error
323	Parameter Error
652	LWM2M session in progress
654	RDMS services are in “deactivated” state
655	RDMS services are in “prohibited” state (see +WDSG command)
656	RDMS services are in “to be provisioned” state; no available NAP
800	SIM Security unspecified error
902	No more sockets available; the maximum number has been reached
903	Memory problem
904	DNS error
905	TCP disconnection by the server
906	TCP/UDP connection error
907	Generic error
908	Fail to accept client request's
909	Data send by KTCPSND/KUDPSND are incoherent
910	Bad session ID
911	Session is already running
912	No more sessions can be used
913	Socket connection timer timeout
914	Control socket connection timer timeout
915	A parameter is not expected
916	A parameter has an invalid range of values
917	A parameter is missing
918	Feature is not supported
919	Feature is not available
920	Protocol is not supported
921	Error due to invalid state of bearer connection

Table 14-2: CME Error Codes (Continued)

<err> Code	Meaning
922	Error due to invalid state of session
923	Error due to invalid state of terminate port data mode
924	Error due to session busy, retry later
925	Failed to decode HTTP header's name, missing ':'
926	Failed to decode HTTP header's value, missing 'cr/lf'
927	HTTP header's name is an empty string
928	HTTP header's value is an empty string
929	Format of input data is invalid
930	Content of input data is invalid or not supported
931	The length of a parameter is invalid
932	The format of a parameter is invalid

Index (AT commands)

Symbols

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z

Index

Symbols

Numerics

A

B

C

D

E

F

G

H

I

L

M

N

O

P

R

S

T

U

V

W

Y